SHARP

ポケットコンピュータ

形名

PC-E200

取扱説明書

保証書付(巻末)
(WITH WARRANTY CARD)

〈はじめに〉

このたびは、シャープポケットコンピュータ〈PC-E200〉をお買い上げいただき、まことにありがとうございました。

この計算機は、くり返し計算や複雑な計算処理に威力を発揮するBASICプログラム機能、電卓と同じように操作して計算を行うことができる関数電卓機能を持ち、またアセンブラ言語CASL（キャッスル）で、プログラム作成、デバッグ、実行を行うことができますので、アセンブラ言語の理解、通産省の情報処理技術者試験の学習などにご利用いただけるポケットコンピュータです。

また、本機はパソコン等でよく使われているＺ80※相当のＣＰＵ（中央演算処理装置）を用いており、機械語モニタ機能を持っています。Ｚ80ＣＰＵに関する書籍は数多く市販されており、Ｚ80の機械語の学習にもご利用いただけます。

正しくお使いいただくために、この取扱説明書をよくお読みください。

なお、この取扱説明書は、「シャープサービス・お客様ご相談窓口一覧表」とともに、必ず保存してください。万一ご使用中にわからないことや不具合が生じたとき、きっとお役に立ちます。

※Ｚ80はザイログ社の登録商標です。

〈ご注意・おねがい〉

- 本機は非常に複雑な機能および組み合わせを有する製品であり、出荷に際して取扱説明書を含め十分なチェックをして万全を期しておりますが、万一ご使用中ご不審な点・お気付きのことがありましたら、もよりのシャープサービス・お客様ご相談窓口までご連絡ください。
  なお運用した結果生じる影響につきましては責任を負いかねますので、あらかじめご了承ください。
- 故障・修理や電池交換の場合は、記憶内容が変化・消失することがあります。別売のプリンタ／カセットインターフェイスとカセットテープレコーダを用いて、記憶内容をカセットテープに記録することをお勧めします。
  記憶内容の保護ならびに損害についての責任は負いかねます。
- 本書に記載のプログラムを使用したことによる金銭上の損害および逸失利益または第三者からのいかなる請求につきましても当社はその責任を負いませんので、あらかじめご了承ください。
- この取扱説明書の内容は改良のため予告なく変更する場合があります。

もくじ

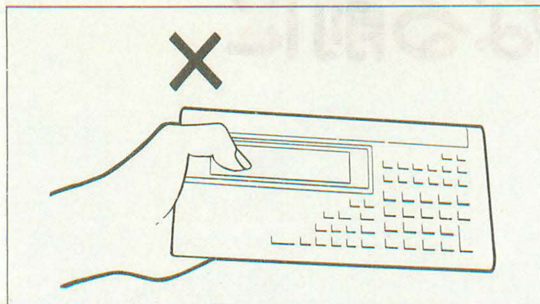
(ページ)

（ページ）

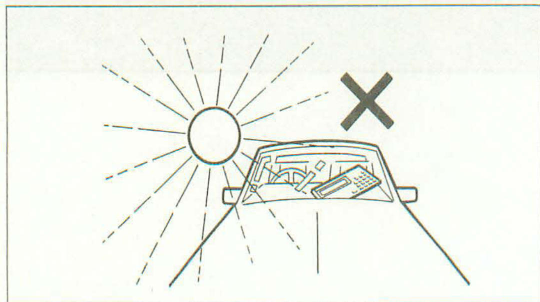
（ページ）

お使いになる前に

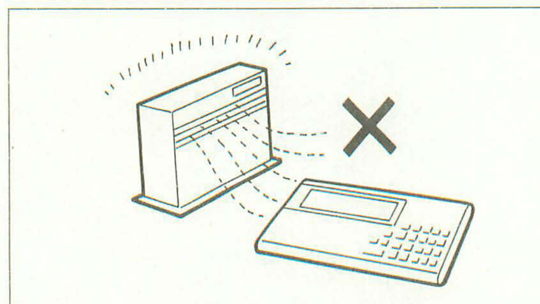
1. おねがい



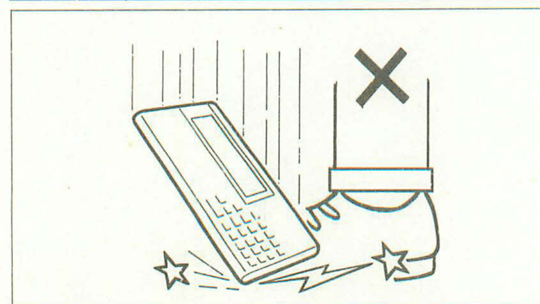
液晶表示部を強く押さえないでください。表示部はガラスですので、押さえると割れることがあります。



窓をしめきった、日の当たる自動車内に放置したり、直射日光が当たる場所に置いたりしないでください。高温により故障の原因になります。



暖房器具の近くなど、高温になる場所に置かないでください。



落としたり、ぶつけたりしないでください。

お手入れに揮発性の液体（シンナー、ベンジンなど）や、ぬれぞうきんなどを使用しないで、乾いた柔らかい布をご使用ください。

この装置は、第二種情報装置(住宅地域又はその隣接した地域において使用されるべき情報装置)で住宅地域での電波障害防止を目的とした情報処理装置等電波障害自主規制協議会(VCCI)基準に適合しております。
しかし、本装置をラジオ、テレビジョン受信機に近接してご使用になると、受信障害の原因となることがあります。
取扱説明書に従って正しい取り扱いをしてください。

正しい取り扱いをしている場合でも、電波の状況によりラジオやテレビの受信に影響を及ぼすことがあります。このようなときには、次の点にご注意ください。

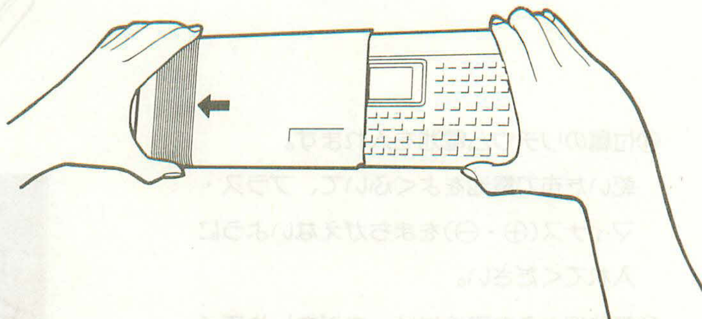
- 本機とラジオ、テレビを十分離してご使用ください。
- 使用されるケーブルは、指定のものをご使用ください。

なお、詳しくはお買い上げの販売店もしくは、もよりのシャープお客様ご相談窓口にお問い合わせください。

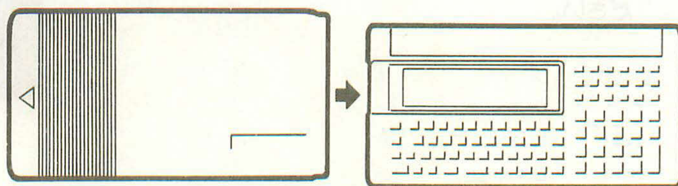
ハードカバーの使いかた

ハードカバーは衝撃に対して計算機を保護する役割を持っています。
計算機を使用しないとき、また、カバンに入れて持ち運ぶときなどにはハードカバーを計算機に取り付けてください。

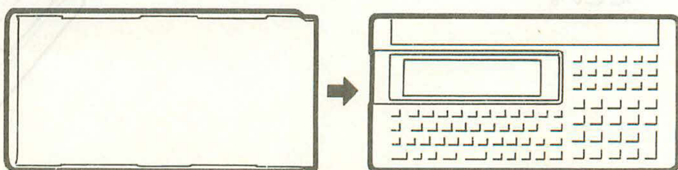
(1)ハードカバーの取りはずしかた



(2)計算機を使わないとき



(3)計算機を使うとき



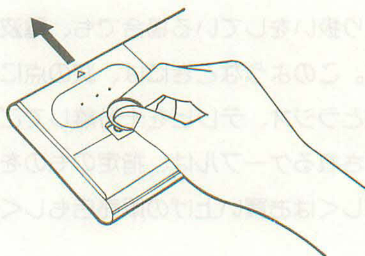
2. お買い上げ直後の操作

(1)電池の取り付け

本機は、本体動作用とメモリ保護用の２種類の電池を入れます。

付属の乾電池４本(動作用)とリチウム電池１個(メモリ保護用)を取り出して、次の手順で入れてください。

①本機裏面の電池ぶた⑬を止めているネジをコイン(貨幣)などでゆるめてください。

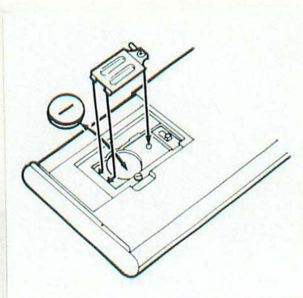


②電池ぶた⑬を図の矢印方向に引いてはずしてください。

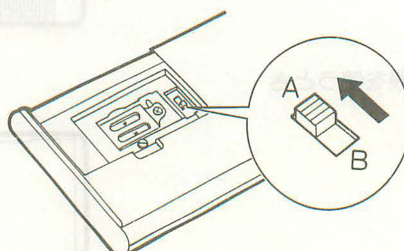
③小型のネジ回しで、メモリ保護用電池の電池押さえのネジをはずし、電池押さえをはずしてください。

④付属のリチウム電池を入れます。
乾いた布で電池をよくふいて、プラス・マイナス(⊕・⊖)をまちがえないように入れてください。

⑤電池押さえを取り付け、ネジをしめてください。



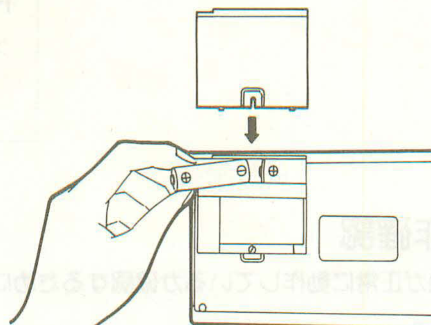
⑥メモリ保護スイッチ⑮をA側に寄せてください。



⑦付属の乾電池をマイナス⊖側から入れてください。⊕・⊖をまちがえないように入れてください。

⑧電池ぶた⑬を取り付け、ネジをしめてください。(ネジはゆるまないようにしめてください。)

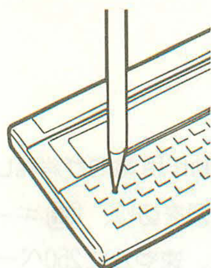
注）電池は4本とも同じ向きに入れてください。



(2)初期設定

電池を入れた直後は、本機の内部が不安定な状態になっています。このため次の操作で初期設定を行ってください。

①ボールペンなどで、本機左端の OFF キーの横にあるリセット(RESET)スイッチ❺を押してください。

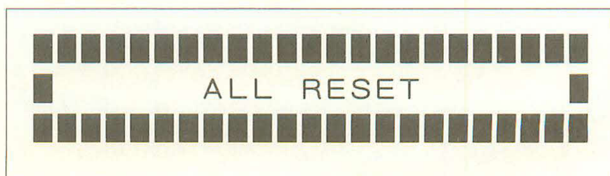


リセットスイッチ❺を操作するときはボールペンなどで押してください。芯先の出たシャープペンシルや先の折れやすいもの、また、針など先のとがったものは使用しないでください。

スイッチを離すと次の画面になります。もし、違う画面になったときはもう一度リセットスイッチ❺を押してください。(画面は"メモリの内容(記憶内容)を消去してもよいか?"ときいています。)

```
MEMORY  CLEAR  O. K. ?  (Y/N)
```

② Y キーを押してください。次の画面（点滅）になります。（初期設定し、記憶内容をすべて消去したことを示しています。）



③どれかキーを押してください。キーを押せば次の画面になります。

```
RUN  MODE
>
```

(3)動作確認

本機が正常に動作しているか確認するために、次のキー操作を行ってください。

[F][R][E][↵]

```
RUN  MODE
FRE
                    3Ø435.
```

以上のように表示すれば本機は正常に動作しており、指令待ち状態になっています。

最後に表示された"3Ø435."という値は、本機にプログラムやデータを入れることができる最大の容量を示しています。

注）以上の(2)～(3)の操作を行っても説明どおりにいかない場合は、もう一度読み直して、操作をやり直してください。

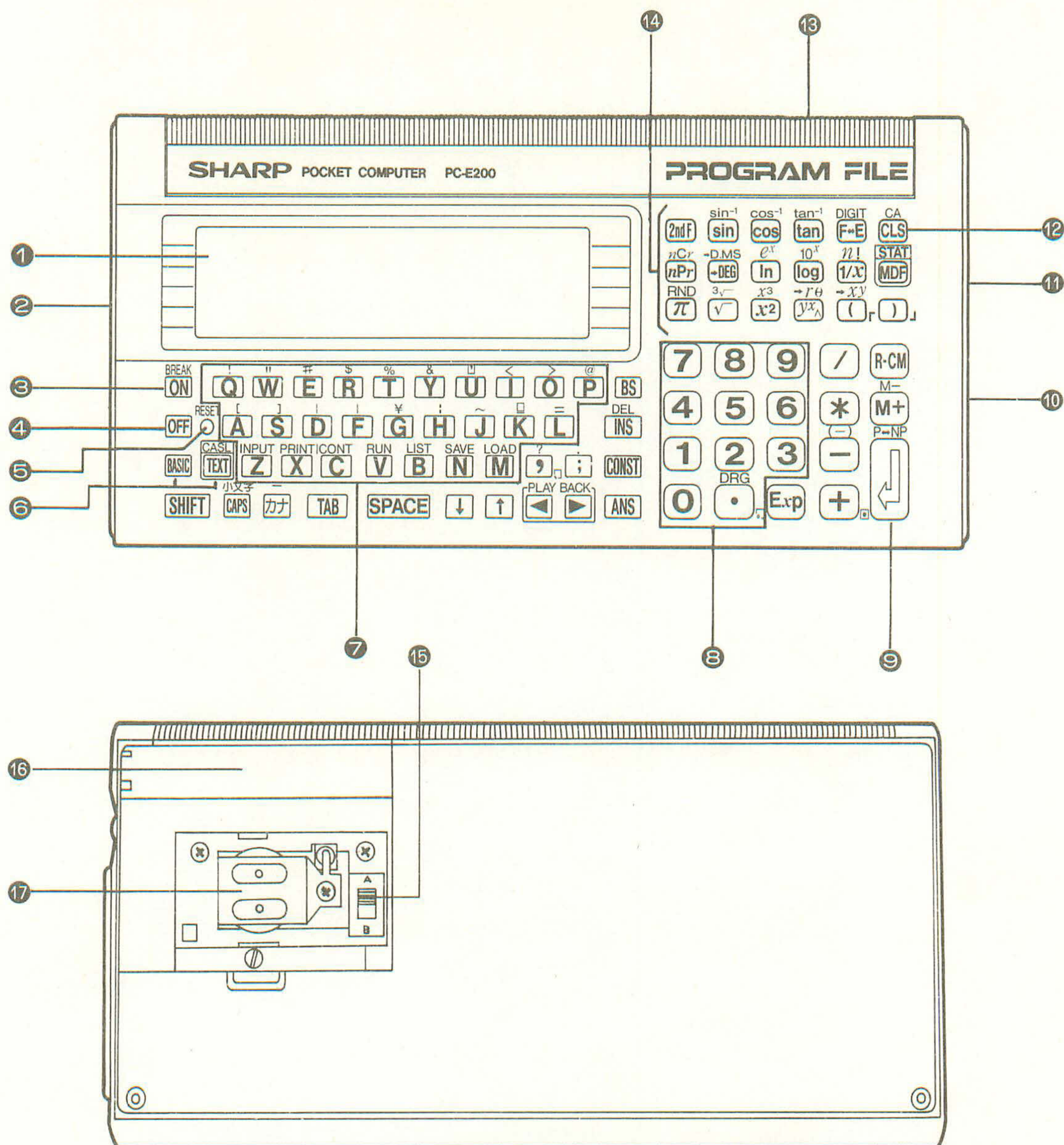
電池の交換について

BATT シンボルが画面右上に点灯したときは、本機内の動作用電池が消耗したことを示しています。このシンボルが点灯したときは、一度 OFF キーで電源を切り、ON キーで再度電源を入れてください。それでもなおこのシンボルが消えないときは、速やかに250ページ以降の説明に従って電池を交換してください。

このシンボルが点灯した状態で使用し、さらに電池が消耗しますと、本機の電源が切れて何も動作しなくなります。

注）電池を交換するとき、別売の周辺機器CE-126Pおよびテープレコーダ(CE-152)をお持ちの場合は、事前にプログラムやデータをテープに記録しておくことをお勧めします。

3. 各部のなまえ



❶表示部
❷周辺機器接続端子(11ピン)
❸電源オン／ブレークキー
❹電源オフキー
❺リセットスイッチ
❻モード切り替えキー
❼アルファベットキー
❽数字キー
❾キャリッジリターンキー
❿システムバス端子
⓫表示濃淡調整つまみ
⓬クリア／クリアオールキー
⓭電池ぶた（裏面）
⓮関数キー
⓯メモリ保護スイッチ
⓰動作用電池収納部
⓱メモリ保護用電池収納部

● 本書の文章等の中で使われている❶～⓱の番号は各部のなまえの番号に対応しています。

基本操作とモードについて

本機には多くのキーがあります。各キーにはいろいろな文字や数字、記号が書かれています。
ここでは、これらの操作のしかた、モード、表示について簡単に説明します。

1．電源のオン／オフ(入り／切り)と表示の濃淡調整

(1)電源のオン／オフ(入り／切り)

本機の左端にある BREAK ON キーを押してください。電源が入って次の表示（画面）になります。

```
      CAPS                    DEG
RUN  MODE
>

    RUN
```

RUNモード（ランモード：計算やプログラムを実行できる状態）になっていることを示します。
＞マークはプロンプト記号といい、本機の準備が完了したことを示しています。

ON キーの下の OFF キーを押せば電源が切れます。

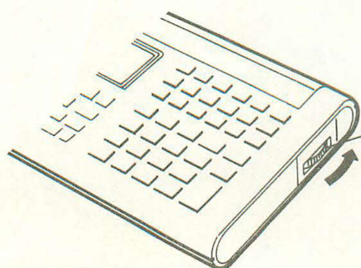
> 電源を入れたり、切ったりしたときに一瞬、画面が黒くなったり、不要な点や線、シンボルが点灯することがありますが、機能などには問題ありません。

●オートパワーオフ機能について

本機は計算実行中（ＢＵＳＹシンボル点灯中）以外の状態で、約11分間新たなキー操作を行わないと、電池の消耗を少なくするため、自動的に電源が切れます。
この場合は ON キーを押せば電源が入ります。

(2)表示の濃淡調整

表示が見やすくなるように表示濃淡調整つまみ⑪を回して濃淡を調整してください。



表示濃淡調整つまみ⑪
矢印の方向へ回せば表示が濃くなり、逆方向へ回せば薄くなります。
表示が見やすくなるように調整してください。

2. モードについて

この計算機は、大きく分けて次の4つのモード(状態)があります。

RUNモード(実行モード)	: BASICプログラムを実行したり、マニュアル操作(手操作)での計算などを行います。
PROモード(プログラムモード)	: BASICプログラムの入力、編集などを行います。
TEXTモード (テキストエディタモード)	: テキストプログラムの入力、編集、削除、またテープ等への記録、読み出し、BASICプログラムとテキストプログラムの変換、CASLのソースプログラムの入力などを行います。
CASLモード(キャッスルモード)	: CASLのソースプログラムのアセンブル、実行を行います。

モードの切り替えは左端にある緑色の[BASIC]と[TEXT]の2つのキーとその下にある[SHIFT]キーで行います。

なお、指定されているモードを示すため、そのモードを表すシンボル(RUN、PRO、CASL、TEXTのどれか)が画面の左下に点灯します。

モードの切り替え

設定するモード	キー 操 作
RUNモード	[BASIC]
PROモード	[BASIC] または [BASIC] [BASIC]
TEXTモード	[TEXT]
CASLモード	[SHIFT] + [CASL]

←RUNモード以外から切り替えるときは[BASIC]キーを2回押す。

● 統計モードについては65ページを参照ください。

[SHIFT] + [CASL] は [SHIFT] キーを押さえたままで [TEXT](CASL) キーを押すことを示します。なお、[2ndF] [CASL] と押しても同じです。

3. 基本的なキー操作

それでは、アルファベットや記号を入れてみましょう。

いったん OFF キーを押して電源を切ってから、改めて ON キーを押して電源を入れてください。

モードはRUNモードになります。

そして、 CLS キーを押して表示内容を消去し、次のようにキーを押してみてください。

なお、キーを押しまちがえたときは、 CLS キーを押してから、改めて最初から操作してください。

(1)アルファベット(大文字)の入れかた

次のキーを押してください。

A B C D E F G

ABCDEFG_

このように、アルファベットの大文字が入ります。

もし、このように押しても、ABCDEFGと入らないときは、次の確認をしてください。

①画面左上に"CAPS"シンボルが点灯していますか。点灯していないときは CAPS キーを押して点灯させてください。

②画面左上に"カナ"シンボルが点灯していませんか。点灯していたら カナ キーを押して消してください。

(2)アルファベット(小文字)の入れかた

次にアルファベットの小文字を入れてみましょう。

CAPS キーを押して画面左上の"CAPS"シンボルを消して、次のキーを押してください。

H I J K L M N

ABCDEFGhijklmn_

このように、"CAPS"シンボルを消して、アルファベットキー❼を押せばアルファベットの小文字が入ります。再び大文字を入力する場合は、"CAPS"シンボルを点灯させてください。

(3)記号の入れかた

次に、＃＄％など、キーの上側に薄茶色で書かれている記号を入れてみましょう。

これら、薄茶色で書かれている記号を入れるときは、次の2つの方法があります。

① SHIFT キーを押さえたまま、これらの記号が書かれているキーを押す。

②これらの記号が書かれているキーを押す前に 2ndF キーを押す。

まず、SHIFT キーを押さえたまま、次のキーを押してください。

E R T (# $ %)

```
ABCDEFGhijklmn#$%_
```

次に、各キーを押す前に 2ndF キーを押します。

2ndF I 2ndF O 2ndF P (< > @)

```
ABCDEFGhijklmn#$%<>@_
```

（2ndF キーを押したとき、画面上側に"2ndF"シンボルが点灯します。）

このように薄茶色で書かれている記号を入れるとき、または薄茶色で書かれている機能を使うときは、SHIFT キーを押さえたまま、それぞれのキーを押すか、それぞれのキーを押す前に、2ndF キーを押します。

次に数字を入れてみましょう。

1 2 3 4 5 6 7 8 9

```
ABCDEFGhijklmn#$%<>@1234
56789_
```

● 1 行24桁を超えると、次の行に入ります。

(4)カーソルについて

次に、◄ キーを1回押してください。すると"9"のところで ▒ マークが点滅します。

```
ABCDEFGhijklmn#$%<>@1234
56789
```

次に ◄ キーを押さえたままにしてみます。

しばらくすると、▒ マークが左に走って行き、左端まで行くと上の行の右端へ移って、また左へ走って行きます。1 行目の左端までくると止まります。

次に ► キーを押さえたままにします。▒ マークが右に走って行き、最後に ▒ マークが消えて _ マークが出ます。

この ▒ マークと _ マークは、カーソルといわれるもので、キーを押したときに文字などが入る位置を示しています。

したがって、もし、キーを押しまちがえて、違う文字や記号が入ってしまったときは、カーソルをまちがった文字などの位置に移動させて、正しいキーを押せば訂正することができます。
(くわしくは29ページを参照)

こんどは [↑] キーを押してみてください。カーソルが1行上(1行目)へ移動します。
次に [↓] キーを押してみてください。カーソルが1行下(2行目)へ移動します。
このように、[↑][↓] キーはカーソルを上下に動かします。
なお、カーソルが上の行へ移動できないときは、その行の先頭(左端)へ移動し、カーソルが下の行へ移動できないときは、その行の最後(文字などの最後)へ移動します。

キーの記載方法について

●この取扱説明書では、キー操作またはキーの機能を示す際、次のように記載します。

CASL / [TEXT] キーの例

①キーに書かれている機能を使用する場合は、単に [TEXT] と記載します。

②キーの上側に薄茶色で書かれている機能を使用する場合は、[SHIFT] + [CASL] または [2ndF] [CASL] と記載します。

なお、[SHIFT] + [CASL] は [SHIFT] キーを押さえたまま [CASL] を押すことを意味します。

また、[2ndF] [CASL] は [2ndF] キーを押して離したあと、[CASL] を押すことを意味します。

●ただし、数字やアルファベット、記号、カタカナなど、単にキーや本体に書かれている文字や数字、記号を入力するだけの場合は、キーの枠や [SHIFT]、[2ndF] キーの操作などを省いて記載します。

例：[S] [H] [A] [R] [P]　→　SHARP

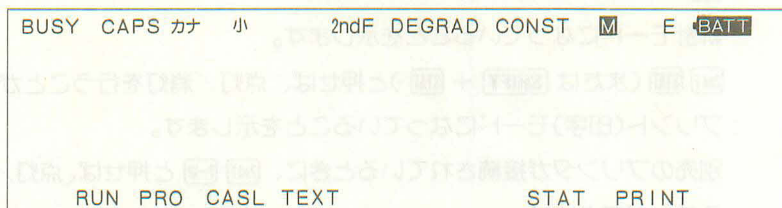
[A] [S] [SHIFT] + [$]　→　AS$

●この計算機は、数字の0をØと表示します。
これは、アルファベットのオー(O)と区別するためで、本書でもオーとゼロの区別がつきにくい場合はゼロをØと表しています。

なお、説明上必要な場合などには、これまで通り [CLS](CA) などの方法で示している場合があります。

4. 表示シンボルについて

本機には色々な機能やモードがあります。そして、いま本機がどのような状態になっているかが分かるように、画面の上や下にシンボルを点灯させます。
次に表示シンボルとその意味を示します。
なお、本書の表示例では、説明上必要なシンボルだけを記載しています。



〈シンボル〉　〈説　明〉

BUSY　：計算機が、計算実行中または命令実行中です。

CAPS　：アルファベットの大文字が入力できる状態になっています。このシンボルが消えているときは、アルファベットの小文字が入力できる状態になっています。この状態は CAPS キーで切り替えます。
ただし、"カナ"シンボルが点灯しているときは、カナの入力が優先され、アルファベットは入力されません。

カナ　：ローマ字のつづりでキーを押せば、カタカナを入力することができます。
カナ キーを押すことにより、このシンボルの点灯／消灯ができます。

小　：小さいカナ文字（アイウエオヤユヨツ）を入力することができます。
"カナ"シンボル点灯時 CAPS キーで、このシンボルを点灯させることができます。

2ndF　：2ndF キーが押されたことを示します。
このシンボルが点灯しているとき、各キーの薄茶色で書かれている機能が指定されていることを示します。

DEG　RAD　GRAD　：三角関数、逆三角関数、座標変換を行うときに使用する角度の単位を示します。（関数計算の項を参照）

CONST　：定数が記憶されていることを示します。（定数計算が設定されていることを示します。39ページ参照）
このシンボルが点灯しているときは、↵ キーで定数計算が行われます。
不要なときは 2ndF CA（SHIFT ＋ CA）と押して、定数を消去してください。

M　：マニュアル（手操作）による計算用のメモリに0以外の数値が入っていることを示します。

E　：エラーが発生したときに点灯します。
エラーは CLS キーを押せば解除されます。

BATT　：本機の動作用電池が消耗したことを示します。このシンボルが点灯したときは、速やかに電池を交換してください。

RUN　：BASICのRUNモード（実行モード）になっていることを示します。
BASIC キーで点灯させせることができます。

PRO　：BASICのPROモード（プログラムモード）になっていることを示します。
BASIC キーで点灯させせることができます。

CASL　：CASLモード（キャッスルモード）になっていることを示します。
SHIFT ＋ CASL と押せば点灯させせることができます。

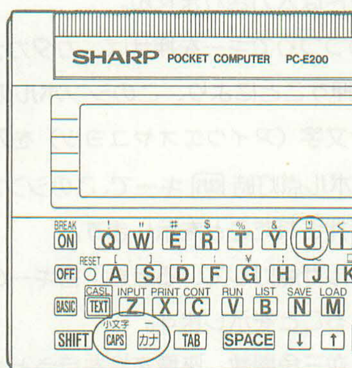
TEXT　：TEXTモード（テキストモード）になっていることを示します。
TEXT キーで点灯させせることができます。

STAT　：統計モードになっていることを示します。
2nd F STAT（または SHIFT ＋ STAT ）と押せば、点灯／消灯を行うことができます。

PRINT　：プリント（印字）モードになっていることを示します。
別売のプリンタが接続されているときに、2nd F P↔NP と押せば、点灯／消灯させることができます。

5. カタカナの入力のしかた

本機はローマ字のつづりで入力した文字をカナに変換する"ローマ字→カナ変換機能"があります。カナは左下にある カナ キーを押して、画面上部に"カナ"シンボルを点灯させ、アルファベットキー❼を用いて入力します。



カナ ……………………このキーで"カナ"シンボルの点灯／消灯を行います。"カナ"が点灯しているときに、カナを入れることができます。

小文字 CAPS ……………………アイウエオヤユヨツ（小さいカナ文字）を入れるときに使います。

SHIFT ＋ U（,） ………「ン」を入れるときに使用します。

〈例〉　（入力文字）　（キー操作）

カタカナ	→	KATAKANA
ガッコウ	→	GAKKOU
ヘンカン	→	HENKAN SHIFT ＋ U（,）
ハンイ	→	HAN SHIFT ＋ U（,） I
ディスク	→	DHISUKU
ウオッチ	→	U 小文字 CAPS OTTI

●「ン」を入れるときに N の後に、Yを除く子音がくる場合は SHIFT ＋ U を押す必要はありません。

●ローマ字の表記の方法については、264ページを参照してください。

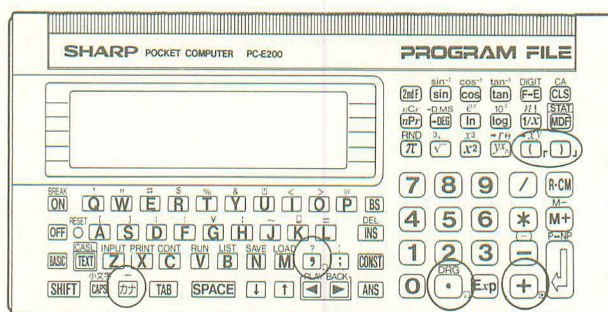
●カナ入力中は、押したキー（子音）を示す表示が画面右に出ます。

〈例〉 SINJY

ジュを入れるためにJYと押したところです。Uを押すとカーソル位置にジュが入り、この表示は消えます。なお、この表示は1行目に出る場合もあります。

【カナ記号の入れかた】

カナ記号は次のキー操作により入力します。



ー（長音符）　：SHIFT ＋ カナ

、（読点）　：, ､

。（句点）　：・ ｡

・（中点）　：＋ ･

「（カギカッコ）：(｢

」（カギカッコ）：) ｣

第１章　マニュアル計算
（手操作による計算）

1.マニュアル計算

マニュアル計算とは、1つ1つの計算を1回1回手操作によって計算することをいいます。関数電卓として使用する場合とほぼ同じですが、BASICプログラムを組む前に、必ず知っておかなければならないことについて説明します。

①計算式の作りかた

②まちがえたときの訂正のしかた

③連続計算のしかた

④関数の使いかた

など、基本的で重要な事項を説明しますので、よく読んで理解し、本機の操作に慣れ、いろいろな基本計算をマスターしてください。

2.マニュアル計算のしかた

①電源オン

ONキーを押して、電源を入れます。

画面の左下に“RUN”が点灯します。これはマニュアル計算ができることを示しています。もし、RUN以外のシンボルが点灯しているときは BASIC キーを押してRUNを点灯させてください。

②計算の記号について

通常の数学では四則計算の記号として「+，−，×，÷」を使いますが、BASIC言語では「×」や「÷」の代りに「*」や「／」などの記号を用います。また、本機ではべき乗や指数、そのほかの関数などにおいて、通常の数学で用いられている記号とは違うかたちで表現されています。以下に本機のマニュアル計算やBASIC言語で使用する計算の記号や特殊記号を示します。

	計　算	数学で用いる記号	本機で用いる記号	読　み　方
(1)	加　算	＋	＋	プラス
(2)	減　算	−	−	マイナス
(3)	乗　算	×	*	アスタリスク
(4)	除　算	÷	／	スラッシュ
(5)	符　号	−または＋	−、＋	マイナス、プラス
(6)	計算の実行	＝	↵	キャリッジリターン

3. キー操作の練習(訂正のしかた)

ここで簡単な例題をもとに、キー操作の練習をします。
数値などを入れているとき、違った数字や命令を入れたり、また余分な数字や命令を入れたり、あるいは抜かしてしまったときの訂正のしかたです。

〔例　題〕①

計　算　式	キ　ー　操　作	答
5+15×2÷4	5[+]15[*]2[/]4[↵]	12.5

①まちがって他のキーを押したとき

5+15×2$\underset{\uparrow}{\div}$4　を　5+15×2$\underset{\uparrow}{\times}$4　とした場合

キ　ー　操　作	表　示　部
5[+]15[*]2[*]4	5+15*2*4_ ↑ここがちがう
[◄][◄]　このキーを押して左へ 2回押す　カーソル※を移動させる	5+15*2*4（*が点滅） ↑この部分にカーソルを移動させて点滅させる
[/] (このあと[↵]としてもよい)	5+15*2/4（4が点滅） 正しいキーを入れる（/）　ここが点滅する（4）
[►]　このキーを押して右へ カーソルを移動させる	5+15*2/4_
[↵]	12.5 答

キー入力した式を確認したいとき(これをプレイバック機能という)

キ　ー　操　作	表　示　部
[◄]　このキーを押すと	5+15*2/4_
[↵]　を押して	12.5
[►]　さらにこのキーを押すと	5+15*2/4（5が点滅） ↑この部分が点滅する
[↵]	12.5

※カーソルについて

キー操作を行うとき表示部に▁が表示されたり、キー操作の訂正のときに▓マークが点滅したりしますが、これらのマークはカーソルと呼ばれるもので、次に操作したキーがこの部分に入ることを示しています。

②まちがって余分なキーを押したとき

〔例　題〕②

計　算　式	キ　ー　操　作	答
20×3÷5	20[＊]3[／]5[↵]	12

キ　ー　操　作	表　示　部
20[＊]3[＊][／]5	20*3*/5_ └余分なキー └数字の0と英文字のOとを区別するために 0(ゼロ)→(オー) 0(ゼロ)と表示する
[◄][◄][◄] このキーを3回押す	20*3*/5 └この部分にカーソルを移動させる
[SHIFT]＋[DEL] （下記の※を参照）	20*3/5 └＊が削除されてここが点滅する
[↵]	12.
キー入力した式を確認したいとき [◄]	20*3/5_

※[SHIFT]＋[DEL]は[SHIFT]キーを押したまま[INS](DEL)キーを押すことを示します。なお、[2nd F][INS](DEL)と押してもDEL機能(削除)が働きます。

③プレイバック機能について

〔例題〕②において、余分なキーを押したのに気がつかずに[↵]キーを押してしまうと、エラーが生じます。このようなとき、キー入力した式を呼びもどして、訂正する機能が、プレイバック機能です。この機能を使うとエラーを起こした場所をカーソルで示しますので、エラーの原因がわかりやすく、たいへん便利です。

キ　ー　操　作	表　示　部
20[＊]3[＊][／]5[↵]	ERROR 10
[◄]	20*3*/5 └ここが点滅をする 例題の式は20×3÷5なので この場合＊が不要です
[BS] （このあと[↵]としてもよい）	20*3/5 └カーソルの前の＊が削除されます。
[►][►]　2回押して	20*3/5_ └カーソルが最後に表示されます。
[↵]	12.

④キー操作が抜けたとき

〔例　題〕③

計算式	キー操作	答
2×3÷10	2[*]3[/]10[↵]	0.6

キー操作	表示部
23[/]10	23/10_ ↑ └ここに* が抜けている
[◄][◄][◄][◄] カーソルを左へ4つ移動	23/10 ↑ └この部分を点滅させます
[INS]	2 3/10 ↑ └こんどはスペースがひとつあいてこの部分が点滅する
[*] (このあと[↵]としてもよい)	2*3/10 ↑ └点滅はここに移ります
[►][►][►][►] カーソルを右へ4つ移動	2*3/10_
[↵]	0. 6

⑤数式において誤りが多いとき

[CLS]キーを押して、今まで入れた内容をすべて消し、あらためて最初から入力し直します。

⑥DELとINS、BSの意味

DEL ―→ DELETE……（デリート）　削除を意味します。

INS ―→ INSERT……（インサート）挿入を意味します。

BS ―→ BACKSPACE……（バックスペース）1文字分の後退を意味します。

⑦計算の優先順位

計算の優先順位の判断や途中結果の処理はすべて計算機が自動的に処理してくれます。計算の優先順位は次の通りです。

1．πや変数の呼び出し
2．関数（SIN、COSなど）
3．べき乗（∧）
4．符号（+、−）
5．乗除算（*、/）
6．加減算（+、−）
7．大小比較（>、>=、<、<=、<>、=）

（注）●カッコが使用されている場合はカッコ内の計算が最優先されます。
●複合関数（sin　$\cos^{-1}$ 0.6など）は右から左の順で計算されます。
●べき乗の連算（3^{4^2} すなわち3∧4∧2など）は右から左の順で計算されます。
●左記3.と4.では後に出てきたほうが優先順位が高くなります。
例）−2∧4は−(2^4)となります。
3∧−2は3^{-2}となります。

4. エラーの処理について

本機で処理できないような計算を行うとエラー(ERROR)になります。

エラーになったときは、次の例題のように処理してください。

〔例　題〕④

$\frac{36}{2+4}$

キ　ー　操　作	表　示　部
36[/]2[+]4[)]	36／2+4)_
[↵]	ERROR 10
[◄]	36／2+4[)] └ここが点滅します この意味は[)](かっこ)の部分がエラーの原因であることを示しています
この〔例題〕④の場合は36／(2+4) が正しい式ですから	
[◄][◄][◄]	36／[2]+4) └ ここが点滅します
[INS]	36／▒2+4)
[(]	36／([2]+4)
[↵]	6.
式を確認したいときは [◄]	36／(2+4)_ └カーソルが最後に表示されます
または [►]	[3]6／(2+4) └ カーソルが先頭に表示されます

5. 数式が長い場合の連続計算のしかた

〔例　題〕⑤

2×3÷10×100
└ここでひとまず区切る

キ ー 操 作	表 示 部
2[*]3[/]10[↵] └→これで区切り計算となります	0.6
[*]100	0.6*100_
[↵]	60.

計算式の途中で[↵]キーを押すと、それまでの数値計算が実行されますので、区切りのよいところで[↵]キーを押し、続いて計算記号と数値をキー入力すれば、長い数値の計算式においても、容易に計算が可能となります。
ただし、式が表示上で255字分以内の長さであれば、区切ることなく計算が可能です。

6. ラストアンサー機能について

これは、計算した結果を記憶している機能です。具体的にはマニュアル計算で[↵]キーを押して得られた結果や、ダイレクトアンサー機能を用いて得られた結果（これらをラストアンサーと呼ぶ）などを記憶します。この機能は、次のような使い方をすると便利です。
ダイレクトアンサー機能：[↵]キーを使用せずに関数計算を行う方法。
数値を入れてから、各関数キーを押す。（42ページ参照）

〔例題〕⑥

$\frac{4\times6}{4+6}=2.4$　この2.4という結果を使って

$\frac{1}{2.4}+\frac{1}{2.4}+\frac{1}{2.4}$ のような計算を行う場合

キー操作	表示部
4[*]6[/][(]4[+]6[)]　[↵]	2.4
1[/][ANS][+]1[/][ANS][+]	1/2.4+1/2.4+_
1[/][ANS]	1/2.4+1/2.4+1/2.4_
[↵]	1.25
ここで[↵]キーを入れたことによって、新しい計算を実行したことになりますのでラストアンサーは、1.25という数値に入れかわります。	

ラストアンサー機能とは、ひとつの記憶箱（レジスタ）にひとつのデータをしまっておくことであり、常に一番新しいデータのみが記憶できます。

さらに、ラストアンサーはRUNモードで使えます。

7.計算結果の表示方法

今まで、四則計算の例を実行していただき本機の取り扱いにもある程度慣れてきたものと思います。ところで、いろいろな計算を実行する中で、小数部については小数点以下2桁、あるいは3桁くらいで十分であるという場合が多いものと思われます。また、小数点の表示ではなく、指数での表示のほうが便利でわかりやすいという場合もあります。

そこで、計算結果の表示方法について、本機の便利な利用法を述べることにします。

まず、われわれの通常用いている数値は、

12300や0.0987のように表されます。

しかし、理科や工科系においては、

12300を　1.23×10^{4}

0.0987を　9.87×10^{-2}

のような表しかたをすることがよくあります。

数値をこのように表す方法を［指数表示方式］と呼びます。

1.23×10^{4}

$\underbrace{9.87}\times\underbrace{10^{-2}}$

この部分を［指数部］といいます。

この部分を［仮数部］といいます。

本機では、数値を指数方式で表示する場合は、指数部を示す記号としてE（Exp キーで入力）を用いて表します。次の例題でキー操作の練習をしてください。

キー操作	表示部
例題 A 1.2×10^{9}= 1 ⊡ 2 Exp 9 ↵ F↔E 普通の表示を指数表示にします	1.2E9 _ 1200000000. 1.2E 09
例題 B 1.2×10^{10}= 1 ⊡ 2 Exp 1 0 ↵ F↔E	1.2E 10 1.2E 10 ここが2桁になりますと F↔E キーを押しても変わりません。
例題 C 0 ⊡ 0987 ↵ F↔E	0.0987 9.87E−02 10^{-2}を意味します
例題 D $\frac{1.71\times2.43\times10^{3}}{3.25\times1.5\times10^{-2}}$= 1 ⊡ 71 * 2 ⊡ 43 Exp 3 / (3 ⊡ 25 * 1 ⊡ 5 Exp −2) ↵ F↔E	85236.92308 8.523692308E 04

（注意）a×10^{x}をa Exp xと置く場合、xは±（0〜99）の整数でなければなりません。もし、整数3桁以上の数値を入れた場合、下から2桁をxと読んでしまいます。また、小数部分を持った数値を入れた場合、エラー10になります。

例　1 Exp −1125 ↵ →1．E−25（誤り）

54ページ 2ndF 10^{x}、2ndF e^{x}、Exp キーのちがいについてを参照

8. 表示に関するフォーマット指定

①DIGIT指定

マニュアル計算で得られた結果などは、特殊計算を除いて小数部は2桁や3桁などでよい場合が実際上多いものです。そこで、計算する前あるいは計算結果が表示されているときにDIGIT（デジット）によって計算結果の小数部の桁数を指定することができます。

〔例〕 2ndF DIGIT	0	整数を表示	（小数点以下1桁目四捨五入）
2ndF DIGIT	2	小数点以下2桁まで表示	（小数点以下3桁目四捨五入）
2ndF DIGIT	4	小数点以下4桁まで表示	（小数点以下5桁目四捨五入）
2ndF DIGIT	・	デジット指定解除	電源オフ、モードの切り替えなどでも解除

〔例 題〕⑦

$\frac{1.71}{3.5}$ の計算を小数点以下2桁まで求めなさい。

キ ー 操 作	表 示 部
まず、デジット指定で小数部の桁数を指定します。 2ndF DIGIT 2 1 ・ 71 / 3 ・ 5 ↵	 0.49

1.71／3.5の結果は0.4885714286ですが小数部分を2桁に指定していますので、3桁目が4捨5入されて0.49となります。

②USING命令

USING（ユージング）命令を用いてもフォーマット指定ができます。この場合は前もって指定しておく必要があります。最後の桁は切り捨てられた値となります。

① USING "###"　符号を含めて整数3桁（符号桁と整数2桁）を表示

② USING "###."　符号を含めて整数3桁と小数点を表示

③ USING "###.##"　符号を含めて整数3桁と小数点と小数点以下2桁を表示

④ USING "###.##∧"　小数以下2桁までの指数方式での表示
このとき仮数部の整数は符号を含めて3桁
指数部は符号を含めて4桁（E－OO）が、
自動的にとられます。

⑤ USING　フォーマット指定を解除

（注意）USING命令を実行すると、デジット指定は解除されます。
同様にデジット指定により、USINGの指定は解除されます。

〔例題〕⑦をUSINGを使用して計算してみましょう。

キー操作	表示部
U S I N G SHIFT+" SHIFT+#	USING "#_
SHIFT+# ・ SHIFT+#	USING "##.#_
SHIFT+# SHIFT+"	USING "##.##"_
↵	>
1・71/3・5↵	0.48 これは「+0.48」ということを意味しています
またここでANSキーを押すと	4.885714286E-01_ USING指定でない表示ができます
そしてさらに↵キーを押すと	0.48 USING指定にもどることができます

※計算が終わったら、U S I N G ↵ と押してUSING指定を解除してください。

9. 数値丸め機能 MDF(モディファイ)

次のような計算を試みてください。

$\frac{55.4}{9}\times9$

キー操作	表示部
(2ndF CA)	
55.4/9↵	6.155555556
*9	6.155555556*9_
↵	55.4

この計算例では表示部に小数点がたくさん出るのですっきりしなく、わずらわしく感じます。そこで有効桁数を考え、式の途中の結果の端数を四捨五入し、しかも、最終結果においてより確からしい値を求める方法があります。これを数値丸め機能(MDF)といいます。

先の例において

55.4÷9=6.155555556ですが、小数第4位を四捨五入し6.156として

6.156×9を計算すれば　55.404となります。

キ　ー　操　作	表　示　部
2ndF DIGIT 3 55.4 / 9 ↵ MDF * 9 ↵	 6.156 6.156 6.156*9_ 55.404

このように数値丸め機能を用いれば、計算途中で有効桁数の端数処理を行いながらより確からしい値を求めることができます。

ただし、この機能(MDF)はDIGITまたはUSINGの指定がされているときのみ有効です。

10. 計算結果の符号の反転

計算によって得られた結果を次の計算に利用するとき、その結果の符号を反転させてから使用したいことがあります。このようなときに便利な機能です。

たとえば、1572円、2350円、3058円の買い物をして10000円で支払うとき、買い物の合計金額とお釣りの金額を求める場合の操作は次のようになります。

キ　ー　操　作	表　示　部
1572 + 2350 + 3058 ↵ 2ndF (−) + 10000 ↵	6980. −6980. 3020.
(注) A−B=−B+Aにより10000−6980を−6980+10000として計算。	
または 1572 + 2350 + 3058 ↵ − 10000 ↵ 2ndF (−)	 6980. −3020. 3020.

11. メモリ計算

マニュアル計算で、数値を一時記憶しておいたり、計算結果を累計したりすることができるメモリがあります。

このメモリは次のキー操作で利用できます。

R·CM ………………メモリの内容を表示します。
続いてもう一回押せばメモリの内容を消去します。メモリ計算は、このキー操作でメモリ内を消去してから始めます。

M+ ………………表示されている数値をメモリに加えます。また、式が表示されているときは、その式を計算して、結果をメモリに加えます。

2ndF M− ……………表示されている数値をメモリから減算します。また、式が表示されているときは、その式を計算して、結果をメモリから減算します。

〔例　題〕⑧

(1)　24×34、64÷5、13×28÷8の計算を行い、その結果の合計を求めなさい。 (2)　次の計算を行いなさい。 　　　24＋46＝ 　　－）37×4＝ 　　＋）84－29＝ 　　　　　　合計

キ　ー　操　作	表　　示　　部	備　　　考
[R·CM][R·CM][CLS]		メモリ内を消去
24[＊]34[M+]	816.	計算結果をメモリに加えます。
64[/]5[M+]	12.8	このとき、メモリが使われていることを示す[M]
13[＊]28[/]8[M+]	45.5	シンボルが点灯します。
[R·CM]	874.3_	合計が求まります。
[R·CM][R·CM][CLS]		メモリ内を消去（[M]シンボル消灯）
24[+]46[M+]	70.	
37[＊]4[2nd F][M−]	148.	37×4の計算を実行し、その結果をメモリから
84[−]29[M+]	55.	減算します。（※参照）
[R·CM]	−23._	合計を呼び出します。

※　次のように操作することもできます。

37[＊]4[↵]　　37×4を計算し、結果を表示

[2nd F][M−]　　表示されている数値をメモリから減算

12.定数計算機能

定数計算とは、ある決まった値(定数)に対して決まった計算を行うことで、次のキーにより定数と計算を設定することができます。

[CONST]………定数と計算を設定

【設定方法】　　（注）aは定数とする数値(または式)を示します。

加算：[+]a[CONST]　および　a[+][CONST]

減算：[−]a[CONST]　および　a[−][CONST]

乗算：[＊]a[CONST]　および　a[＊][CONST]

除算：[/]a[CONST]　および　a[/][CONST]

定数計算を設定すると表示部右上にCONSTシンボルが点灯します。

【設定内容の確認】

CONSTシンボルが点灯しているときに次のキーを押せば設定内容が表示されます。

[2nd F][CONST]　（[SHIFT]＋[CONST]）

【設定内容の消去】

次のキー操作で設定内容が消去されます。また、電源を切ったときも消去されます。

2ndF CA（SHIFT＋CA）………CONSTシンボルが消灯します。

〔例　題〕⑨

(1)　＋(4.8+3.6)を定数とする、次の計算を行いなさい。
32.5+(4.8+3.6)=
24−18.5+(4.8+3.6)=
8.2×6+(4.8+3.6)=

(2)　(57−14)×を定数とする、次の計算を行いなさい。
(57−14)×18=
(57−14)×(27／4)=
(57−14)×(24+43)=

キー操作	表示部	備考
＋ (4.8 ＋ 3.6)	+(4.8+3.6)_	
CONST	>	定数計算設定（CONSTシンボル点灯）
32.5 ↵	40.9	
24 − 18.5 ↵	13.9	
8.2 ＊ 6 ↵	57.6	
2ndF CONST	+8.4_	設定内容確認（4.8+3.6の結果が定数になっています）
(57 − 14) ＊	(57−14)*_	
CONST	>	定数計算設定（前の設定内容は消されます。）
18 ↵	774.	
27 / 4 ↵	290.25	
24 ＋ 43 ↵	2881.	
2ndF CA	>	設定内容消去

第2章

マニュアル計算における関数計算の操作方法と練習

1. マニュアルにおける関数計算

マニュアルにおいて、関数計算を単独で行う場合の方法をここで説明します。
本機には、BASIC 言語でも使える基本関数が多くありますが、それらについてBASICプログラムの項で説明します。

①関数キーを使う方法

マニュアル計算の場合は、ふつう、 **関数キー** を使います。一般的にはこの方法が速いでしょう。

> **RUN　モード**
> ① **関数キー** を入れ、続いて **数値** [↵] キーを入力する。
> ② **数値** を入れ、続いて **関数キー** を操作する。
> この場合、関数電卓と同じ使いかたとなり、[↵] キーを押さなくても計算結果が出ます。これをダイレクトアンサー機能といいます。ただし、この機能は、べき乗や座標変換のように2つの数値が必要な関数、また式の入力途中では働きません。
> 表の中にこのダイレクトアンサー機能を(DAns機能)と表示してあります。

②アルファベットキーを使う方法

関数キー を用いなくても、アルファベットで表現されるものは、アルファベットキーで入力することができます。関数キーが用意されていない関数もアルファベットキーで入力します。
たとえば
5^2の計算では、[x^2] 5と操作しても、[S][Q][U] 5と操作しても答は同じです。プログラムの中で関数を扱う場合は、アルファベットの使用頻度が高いため、この方法が便利なことがあります。

①関数入力の方法 1

（注）DAns：ダイレクトアンサー

項　目	通常の表現	本機の表現	読 み か た	入力方法と備考
2　乗	x^2	SQU	スクウェア square （平方）	[x²] X [↵] または X [x²]（DAns機能）
平方根	$\sqrt{x}$	SQR	ルート root square root （平方根）	[√] X [↵] または X [√]（DAns機能）
3　乗	x^3	CUB	キュービック cubic （立方）	[2nd F] [x³] X [↵] または X [2nd F] [x³]（DAns機能）
立方根	$\sqrt[3]{x}$	CUR	キュービック ルート cubic root （立方根）	[2nd F] [∛] X [↵] または X [2nd F] [∛]（DAns機能）
逆　数	$\frac{1}{x}$	RCP	レシプロカル reciprocal （逆数）	[1/x] X [↵] または X [1/x]（DAns機能）
円周率	π	PI	パイ P*i* （ギリシャ語・円周率）	[π] [↵] または [P] [I] [↵] $\pi \fallingdotseq 3.141592654$
べき乗 （べき乗根）	y^x （注） $\sqrt[x]{y}=y^{\frac{1}{x}}$	∧	ハットマーク power （累乗，べき）	Y [yˣ] X [↵] （注）Y [yˣ] [(] 1 [/] X [)] [↵]
絶対値	$\|x\|$	ABS	アブソリュート absolute （絶対値）	[A] [B] [S] X [↵] Xの絶対値

（注）「入力方法と備考」の欄のX、Y、n、rは「通常の表現」欄のx、y、n、rに対応する数値を表します。また、θは角度を表す数値です。

②関数入力の方法[2]

項　目	通常の表現	本機の表現	読みかた	入力方法と備考
整数化		INT	インテジャー integer （整　数）	[I][N][T] X [↵] X以下でかつ最も大きい整数
符号関数		SGN	シグナム sign （符　号）	[S][G][N] X [↵] Xの値が正のとき1，負のとき−1, 0のとき0が得られる
乱　数		RND	ランダム randum numbers （乱　数）	[2ndF][RND] X [↵] 乱数を発生する ● Xについては 97 ページを参照ください。
数値丸め機能		MDF	モディファイ modify （修正する）	[MDF] X [↵] DIGITまたはUSING命令で指定された桁数の端数処理を行う。
桁数指定		DIGIT	デジット digit	[2ndF][DIGIT] n ただし、nは小数点以下の桁数を指定 0≦n≦9 解除は[2ndF][DIGIT][・]とする。
階　乗	n!	FACT	ファクトリアル factorial （階　乗）	[2ndF][n!] n [↵] または n [2ndF][n!]（DAns機能）
順　列	$nPr=\frac{n!}{(n-r)!}$	NPR	パーミュテーション permutation （順　列）	[nPr][(] n [,] r [)][↵] ただし、n≧r
組合せ	$nCr=\frac{n!}{r!(n-r)!}$	NCR	コンビネーション combination （組合せ）	[2ndF][nCr][(] n [,] r [)] [↵] ただし、n≧r
対数関数	ln x または $\log_e x$	LN	ローン natural logarithm （自然対数）	[ln] X [↵] または X [ln]（DAns機能）
	log x	LOG	ロガリズム common logarithm （常用対数）	[log] X [↵] または X [log]（DAns機能）
指数関数	e^x （自然指数）	EXP	エクスポーネント exponent （指数）	[2ndF][e^x] X [↵] または X [2ndF][e^x]（DAns機能） e＝2.718281828

③関数入力の方法 3

項目	通常の表現	本機の表現	読みかた	入力方法と備考
指数関数	10^x （常用指数）	TEN	テン power of ten	[2nd F][10^x] X [↵] または X [2nd F][10^x]（DAns機能）
三角関数	$\sin\theta$	SIN	サイン sine （正弦）	●[2nd F][DRG]でDEG、RAD、GRAD、のいずれかを指定 [sin] θ [↵] または θ [sin]（DAns機能）
	$\cos\theta$	COS	コサイン cosine（余弦）	[cos] θ [↵] または θ [cos]（DAns機能）
	$\tan\theta$	TAN	タンジェント tangent（正接）	[tan] θ [↵] または θ [tan]（DAns機能）
逆三角関数	$\sin^{-1}x$	ASN	アークサイン arc sine （逆正弦）	●[2nd F][DRG]でDEG、RAD、GRAD、のいずれかを指定 [2nd F][sin⁻¹] X [↵] または X [2nd F][sin⁻¹]（DAns機能）
	$\cos^{-1}x$	ACS	アークコサイン arc cosine （逆余弦）	[2nd F][cos⁻¹] X [↵] または X [2nd F][cos⁻¹]（DAns機能）
	$\tan^{-1}x$	ATN	アークタンジェント arc tangent （逆正接）	[2nd F][tan⁻¹] X [↵] または X [2nd F][tan⁻¹]（DAns機能）

項　目	通常の表現	本機の表現	読みかた	入力方法と備考	
双曲線関数	sinh$x$	HSN	ハイパボリックサイン hyperbolic sine	[H][S][N] X [↵] $\frac{e^x-e^{-x}}{2}$	詳細については専門書参照のこと
	cosh$x$	HCS	ハイパボリックコサイン hyperbolic cosine	[H][C][S] X [↵] $\frac{e^x+e^{-x}}{2}$	
	tanh$x$	HTN	ハイパボリックタンジェント hyperbolic tangent	[H][T][N] X [↵] $\frac{e^x-e^{-x}}{e^x+e^{-x}}$	
逆双曲線関数	$\sinh^{-1}x$	AHS	アークハイパボリックサイン arc hyperbolic sine	[A][H][S] X [↵] $\log_e(x+\sqrt{x^2+1})$ xは任意の数	
	$\cosh^{-1}x$	AHC	アークハイパボリックコサイン arc hyperbolic cosine	[A][H][C] X [↵] $\pm\log_e(x+\sqrt{x^2-1})$ $x>1$	
	$\tanh^{-1}x$	AHT	アークハイパボリックタンジェント arc hyperbolic tangent	[A][H][T] X [↵] $\log_e\sqrt{\frac{1+x}{1-x}}$　$-1<x<1$	

④関数入力の方法 4

項目　通常の表現	本機の表現	読みかた	入力方法と備考
60進数(度,分,秒)→10進数(度)の変換	DEG	ディグリー to decimal degree	→DEG X ↵ またはX →DEG（DAns機能）
10進数(度)→60進数(度,分,秒)の変換	DMS	ディ・エム・エス to degrees, minutes, seconds	2ndF →D.MS X ↵ またはX 2ndF →D.MS（DAns機能）
16進数→10進数の変換	&H	アンド・ヘキサデシマル & hexadecimal	2ndF & H h ↵ ただし、hは0～FFFFFFFFまでの16進数（A～Fはアルファベットキーで入力）
直交座標→極座標の変換 $Z=X+Yi$ → $Z=(r,\theta)$	POL	ポーラ to polar cordinates（極）	●DEG、RAD、GRADのいずれかを指定 2ndF →rθ (X , Y) ↵→rの値 Z ↵→θの値 ここで、rを確認したいときは Y ↵→rの値 さらに、ここでθを確認したいときは Z ↵→θの値
極座標→直交座標の変換 $Z=(r,\theta)$ → $Z=X+Yi$	REC	レキュタンギュラー to rectangular cordinates（直角の,長方形）	●DEG、RAD、GRADのいずれかを指定 2ndF →xy (r , θ) ↵→Xの値 Z ↵→Yの値 ここでXの値を確認したいときは Y ↵ さらにここでYの値を確認したいときは Z ↵ （注意）座標変換では、計算結果を変数ZとYに入れますので、それまでZ、Y(あるいはZ$、Y$)に入っていた内容は消されます。

⑤取り扱う角度の単位の指定について

三角関数、逆三角関数および座標変換などの計算では、取り扱う角度の単位を正しく指定しておく必要があります。

マニュアル計算では、2ndF DRG の操作で指定することができますので、度・ラディアン・グラードの指定をしたうえで計算を行ってください。

2ndF DRG の操作をしなくても、次の命令で指定することができます。

角度単位	命令	表示シンボル	備考
度	DEGREE ↵	DEG	直角を90で表わす単位〔°〕
ラディアン	RADIAN ↵	RAD	直角を$\frac{\pi}{2}$で表わす単位〔rad〕
グラード	GRAD ↵	GRAD	直角を100で表わす単位〔g〕

これは、BASIC言語でプログラムを組んでいくときなどに使いますので、よく使いなれておいてください。

2. マニュアル計算における関数計算の操作方法と練習

1 2乗 [x^2]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例	題	キー操作	表示部	答
1	37^2	[x^2]37 [↵]	SQU 37_ 1369.	 1369
		または 37[x^2]	 1369.	 1369
2	$(87\times57)^2$	[x^2][(]87[*]57[)] [↵]	SQU（87*57)_ 24591681.	 24591681
		または 87[*]57[↵][x^2]	 24591681.	 24591681
3	$\sqrt{5^2-4^2}$	[√][(][x^2]5[−] [x^2]4[)] [↵]	SQR(SQU5−SQU4)_ 3.	 3
		または 5[x^2][−][x^2]4 [↵] [√]	 9. 3.	 3

2 平方根 [√]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例	題	キー操作	表示部	答
1	$\sqrt{3}$	[√]3 [↵]	SQR3_ 1.732050808	 1.732050808
		または 3[√]	 1.732050808	 1.732050808
2	$\sqrt{25+86}\times\sqrt{37}$	[√][(]25[+]86[)] [*][√]37 [↵]	SQR(25+86)* SQR37_ 64.08587988	 64.08587988
		または 25[+]86[↵][√][*] [√]37 [↵]	10.53565375*_ 10.53565375*SQR37_ 64.08587986	 64.08587986

③ 3乗 [2nd F] [x^3]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例題		キー操作	表示部	答
1	3^3	[2nd F][x^3]3 [↵] または 3[2nd F][x^3]	CUB 3_ 27. 27.	 27 27
2	2^3+3^3	[2nd F][x^3]2[+][2nd F][x^3]3 [↵] または 2[2nd F][x^3] [+][2nd F][x^3]3 [↵]	CUB 2+CUB 3_ 35. 8. 8.+CUB 3_ 35.	 35 35

④ 立方根 [2nd F] [$\sqrt[3]{\ }$]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例題		キー操作	表示部	答
1	$\sqrt[3]{125}$	[2nd F][$\sqrt[3]{\ }$]125 [↵] または 125[2nd F][$\sqrt[3]{\ }$]	CUR 125_ 5. 5.	 5 5
2	$\sqrt[3]{(25+38)\times(96-57)}$	[2nd F][$\sqrt[3]{\ }$][(][(]25[+]38[)] [*][(]96[−]57[)][)] [↵] または 25[+]38[↵] [*][(]96[−]57[)] [↵] [2nd F][$\sqrt[3]{\ }$]	CUR((25+38)* (96−57))_ 13.49382434 63. 63.*(96−57)_ 2457. 13.49382434	 13.49382434 13.49382434

5 逆数 [1/X]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例題		キー操作	表示部	答
1	$\frac{1}{125}$	[1/X]125 [↵] または 125[1/X]	RCP 125_ 0.008 0.008	0.008 0.008
2	$\frac{1}{\sqrt{185}}$	[1/X][(][√]185[)] [↵] または 185[√] [1/X]	RCP (SQR185)_ 0.073521462 13.60147051 0.073521462	0.073521462 0.073521462
3	$\frac{1}{\frac{1}{6}+\frac{1}{7}}$	[1/X][(][1/X]6[+][1/X]7[)] [↵] または 6[1/X][+][1/X]7[↵] [1/X]	RCP(RCP6+ RCP7)_ 3.230769231 0.309523809 3.23076923	3.230769231 3.23076923

6 べき乗 [y^x ∧]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例題		キー操作	表示部	答
1	8^5	8[y^x]5 [↵]	8∧5_ 32768.	32768
2	$8^{5.37}$	8[y^x]5.37 [↵]	8∧5.37_ 70728.30171	70728.30171
3	5×7^4	5[*]7[y^x]4 [↵]	5*7∧4_ 12005.	12005
4	$2.7^{3.4}+4.3^{2.5}$	2.7[y^x]3.4[+]4.3 [y^x]2.5 [↵]	2.7∧3.4+4.3∧2.5_ 67.62611161	67.62611161
5	4^{-2} $\left(\frac{1}{4^2}\right)$	4[y^x][−]2 [↵]	4∧−2_ 0.0625	0.0625

7 べき乗根 [y^x]

平方根 $\sqrt{\ }$（$\frac{1}{2}$乗），立方根 $\sqrt[3]{\ }$（$\frac{1}{3}$乗）を一般化すると、

$\sqrt[x]{Y}=Y^{\frac{1}{x}}$

の関係になります。つまりこれは、べき乗の変形と考えられます。

①べき乗根の公式

m，n，pは正の整数，a>0，b>0とするとき，以下の式がなりたちます。

1．$\sqrt[n]{a}=a^{\frac{1}{n}}$

2．$(\sqrt[n]{a})^n=a$

3．$(\sqrt[n]{a})^m=\sqrt[n]{a^m}=a^{\frac{m}{n}}$

4．$\sqrt[np]{a^{mp}}=a^{\frac{mp}{np}}=a^{\frac{m}{n}}$

5．$\sqrt[n]{\sqrt[m]{a}}=(\sqrt[n]{a^{\frac{1}{m}}})=a^{\frac{1}{mn}}=\sqrt[mn]{a}$

6．$\sqrt[n]{a}\cdot\sqrt[n]{b}=\sqrt[n]{a\cdot b}$

7．$\frac{\sqrt[n]{a}}{\sqrt[n]{b}}=\sqrt[n]{\frac{a}{b}}$

8 階　乗 [2nd F][n!]
順　列 [nPr]　組合せ [2nd F][nCr]

①階乗

nから１までの整数をかけ合せてつくった値をn！と表し、nの階乗という。

n！＝n×(n-1)×(n-2)×………×3×2×1

たとえば

5！＝5×4×3×2×1＝120

このようになります。

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例	題	キ ー 操 作	表 示 部	答
1	5！	[2nd F][n!]5 [↵] または 5[2nd F][n!]	FACT5_ 120. 120.	120 120
2	8!×5!	[2nd F][n!]8[*][2nd F][n!]5 [↵]	FACT8*FACT5_ 4838400.	4838400

②順列・組合せ

相異なるn個のものからr個をとって1組としたものを、n個のものからr個をとる組合せ(Combination)といい、その仕方の数をnCrで表します。

r個をとり出した上で、さらにとり出したr個のものを1列に順序づけて並べたものを、n個のものからr個をとる順列(Permutation)といい、その仕方の数をnPrで表します。

順　列　$_{n}P_{r}=\dfrac{n!}{(n-r)!}$　　(r≦n)

組合せ　$_{n}C_{r}=\dfrac{n!}{r!(n-r)!}$　　ここで　0！＝1と定義します。

例　　題		キ　ー　操　作	答
1	$_{10}P_{7}=\dfrac{10!}{(10-7)!}$	[nPr][(]10[,]7[)][↵] または [2ndF][n!]10[/][2ndF][n!][(]10[−]7[)] [↵]	604800 604800
2	$_{10}C_{4}=\dfrac{10!}{4!(10-4)!}$	[2ndF][nCr][(]10[,]4[)][↵] または [2ndF][n!]10[/][(][2ndF][n!]4[*] [2ndF][n!]6[)] [↵]	210 210

9 常用対数 [log]

$a^m=M$のとき、指数mはaを底とするMの対数といい、$m=\log_a M$と表します。

このときMを対数mの真数といいます。

真数は常に正です。とくに、底が10の対数を常用対数と呼び、底を略して log Mと表します。

対数の基本的性質

1. $\log 1=0$
2. $\log 10=1$
3. $\log 100=2$
4. $\log M\cdot N=\log M+\log N$　(対数で表すと、掛け算がたし算になります)
5. $\log \dfrac{M}{N}=\log M-\log N$　(対数で表すと、わり算が引き算になります)
6. $\log M^P=P\cdot\log M$
7. $\log \sqrt[n]{M^m}=\log M^{\frac{m}{n}}=\dfrac{m}{n}\log M$
8. $\log_a b=\dfrac{\log b}{\log a}$
9. $\log_b a=\dfrac{1}{\log_a b}$ (底の変換公式)
10. $10^{\log M}=M$

例題をはじめる前にCLSキーで表示をクリアしてください。

例	題	キ ー 操 作	表 示 部	答
1	log1000	log 1000 ↵ または 1000 log	LOG1000– 3. 3.	3 3
2	$\log\sqrt{15} = \frac{1}{2}\log 15$	1 / 2 * log 15 ↵	1/2*LOG15_ 0.588045629	0.588045629
3	$\log\frac{9}{15}$	log (9 / 15) ↵	LOG(9/15)_ –0.221848749	–0.221848749

10 常用指数 2ndF 10ˣ

10のべき乗を求めるときに用います。

これは $\log x$ の逆関数で、対数の真数を求めることになります。（59ページ逆三角関数の項参照）

例題をはじめる前にCLSキーで表示をクリアしてください。

例	題	キ ー 操 作	表 示 部	答
1	10^3	2ndF 10ˣ 3 ↵	TEN3_ 1000.	1000
2	$10^{7.4}+10^{8.3}$	2ndF 10ˣ 7.4 + 2ndF 10ˣ 8.3 ↵	TEN7.4+TEN8.3_ 224645095.8	224645095.8
3	$\log N=1.5$ ⬇ $N=10^{1.5}$	2ndF 10ˣ 1.5 ↵	TEN1.5_ 31.6227766	31.6227766

11 自然対数 ln

10を底とする対数が常用対数であるのに対し、

$e=1+\frac{1}{1!}+\frac{1}{2!}+\frac{1}{3!}+\cdots=2.718281828459\cdots\cdots$

を底とする対数を、自然対数またはネーピアの対数といいます。

自然対数と常用対数の関係

$\log x=\frac{\ln x}{\ln 10}=0.434294481\times\ln x$

例題をはじめる前にCLSキーで表示をクリアしてください。

例	題	キー操作	表示部	答
1	ln10	ln 10 ↵ または 10 ln ↵	LN10_ 2.302585093 2.302585093	2.302585093 2.302585093
2	ln18+ ln15	ln 18 + ln 15 ↵	LN18+LN15_ 5.598421959	5.598421959
3	ln(18×15)	ln (18 * 15) ↵	LN(18*15)_ 5.598421959	5.598421959
4	ln16− ln9	ln 16 − ln 9 ↵	LN16−LN9_ 0.575364144	0.575364144
5	$\ln\left(\frac{16}{9}\right)$	ln (16 / 9) ↵	LN(16／9)_ 0.575364144	0.575364144

⑫ 自然指数 2ndF e^x

この関数は ln$x$の逆関数（真数）を求める関数です。（逆三角関数の項参照）

例題をはじめる前にCLSキーで表示をクリアしてください。

例	題	キー操作	表示部	答
1	e^1	2ndF e^x 1 ↵ または 1 2ndF e^x ↵	EXP1_ 2.718281828 2.718281828	2.718281828 2.718281828
2	$e^{2.845}$	2ndF e^x 2.845 ↵	EXP2.845_ 17.20155867	17.20155867
3	$e^{-6.41}\times e^{8.65}$	2ndF e^x − 6.41 * 2ndF e^x 8.65 ↵	EXP−6.41* EXP8.65_ 9.393331288	9.393331288

2ndF 10^x、2ndF e^x、Exp キーのちがいについて

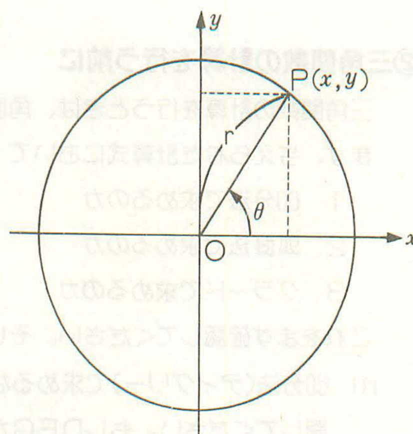
2ndF 10^x	常用指数	10のべき乗を求めます	数式例 ① $10^{1.5}$ ② $10^{7.4}$ ③ $10^{-1.3\times6.4}$

2ndF e^x	自然指数	自然数eのべき乗を求めます	数式例 ① e^1 ② $e^{-2.5}$ ③ $e^3 \times e^{8.5}$
Exp	指数指定	指数部を置数するときに使います 指数は±(0〜99)の整数です。 35ページ参照	置数例 ① 2.5×10^3 2.5 Exp 3と置数 (2.5E3と表示されます) ② 2×10^{-3} 2 Exp −3と置数 (2E−3と表示されます)

13 三角関数 sin cos tan

正弦(sinθ)、余弦(cosθ)、正接(tanθ)を求めます。

右の図のように、xy平面上で、動径OPが始線Ox(x軸の正方向)となす角をθとし、動径上の点Pの座標を(x, y)、$\overline{OP}=\sqrt{x^2+y^2}=r$とすると、次のような関係がなりたちます。



1. sin$\theta=\frac{y}{r}$ (正弦) { in$\theta=\frac{y}{r}$ }
2. cos$\theta=\frac{x}{r}$ (余弦) { os$\theta=\frac{x}{r}$ } ―このように考えるとわかりやすい。
3. tan$\theta=\frac{y}{x}$ (正接) { an$\theta=\frac{y}{x}$ }

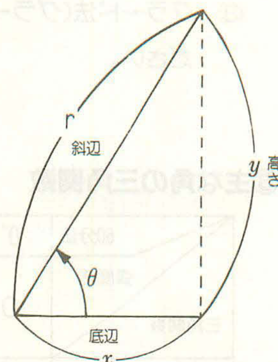
sinは$\frac{高さ}{斜辺}$で求められますから

Sを筆記体で書くと$\mathcal{S}$となりますので

斜辺 高さ で$\frac{y}{r}$になるという意味です。

Cosの場合もCを∠と考えて 斜辺 底辺 で$\frac{x}{r}$

Tanの場合もTを⅃と考えて 高さ 底辺 で$\frac{y}{x}$



①角度の単位

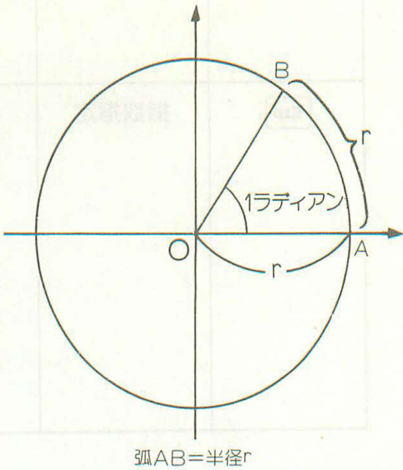
半径に等しい長さの弧に対する中心角は、円の大きさに関係なく一定です。

この一定の角を単位1(ラディアン)として、角を測る方法が弧度法です。

弧度法では、単位の名称を略して、角の大きさを無名数で表すのがふつうです。

弧度法に対して、度を単位として角を測る方法(1直角=90°, 1°=60′, 1′=60″)を60分法といいます。

また、この90°を100^g(グラード)として測る方法もあります。



弧AB=半径r

60分法	弧度法	グラード法
ディグリー	ラディアン	グラード
DEG	RAD	GRAD
90° ⟷	$\frac{\pi}{2}$ ⟷	100^g
180° ⟷	π ⟷	200^g
360° ⟷	2π ⟷	400^g

②三角関数の計算を行う前に

三角関数の計算を行うときは、角度単位の指定をしてください。

まず、与えられた計算式において

1. 60分法で求めるのか
2. 弧度法で求めるのか
3. グラードで求めるのか

これをまず確認してください。そして、

(1) 60分法(ディグリー)で求めるなら、表示部の上側に小さな英字でDEGが出ていることを確認してください。もしDEGが出ていないならDEGという英字が出るまで 2ndF DRG をくり返してください。(DEG表示は、電源ON時に必ず表示されます。)

(2) 弧度法(ラディアン)で求めるなら、表示部にRADが出るまで 2ndF DRG をくり返してください。

(3) グラード法(グラード)で求めるなら、表示部にGRADが出るまで 2ndF DRG をくり返してください。

③主な角の三角関数

60分法	0°	30°	45°	60°	90°	120°	135°	150°	180°	270°	360°
弧度法 / 三角関数	0	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$	$\frac{2\pi}{3}$	$\frac{3\pi}{4}$	$\frac{5\pi}{6}$	π	$\frac{3\pi}{2}$	2π
$\sin\theta$	0	$\frac{1}{2}$	$\frac{1}{\sqrt{2}}$	$\frac{\sqrt{3}}{2}$	1	$\frac{\sqrt{3}}{2}$	$\frac{1}{\sqrt{2}}$	$\frac{1}{2}$	0	−1	0
$\cos\theta$	1	$\frac{\sqrt{3}}{2}$	$\frac{1}{\sqrt{2}}$	$\frac{1}{2}$	0	$-\frac{1}{2}$	$-\frac{1}{\sqrt{2}}$	$-\frac{\sqrt{3}}{2}$	−1	0	1

tanθ	0	$\frac{1}{\sqrt{3}}$	1	$\sqrt{3}$	∞	$-\sqrt{3}$	−1	$-\frac{1}{\sqrt{3}}$	0	∞	0
図による角度と符号											

1 SIN関数

例題をはじめる前にCLSキーを押して表示をクリアしてください。

	例　題	キ　ー　操　作	表　示　部	答
1	sin60°	表示部にDEGを表示させます sin 60 ↵ または 60 sin	SIN60_ 0.866025403 0.866025403	 0.866025403 0.866025403
2	sin$\frac{\pi}{3}$	表示部にRADを表示させます sin (π / 3) ↵ または π / 3 ↵ sin	SIN(PI／3)_ 0.866025403 0.866025403	 0.866025403 0.866025403
3	sin(0.5π+2.4)	表示部にRADを表示させます sin (0.5 * π + 2.4) ↵	 SIN(0.5*PI+2.4)_ −0.737393715	 −0.737393715
4	sin²30°	表示部にDEGを表示させます sin 30 y^x 2 ↵ または 30 sin y^x 2 ↵	SIN30^2_ 0.25 0.25	 0.25 0.25
5	sin18°30′	表示部にDEGを表示させます sin →DEG 18.30 ↵ または →DEG 18.30 ↵ sin または 18.30 →DEG sin	SIN DEG18.30_ 0.317304656 0.317304656 0.317304656	 0.317304656 0.317304656 0.317304656

6	$\sin 250^g$	表示部にGRADを表示させます [sin]250 [↵]	SIN250_ −0.707106781	−0.707106781
		または 250[sin]	−0.707106781	−0.707106781

② COS関数

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例	題	キー操作	表示部	答
1	$4\times\cos 25^\circ$	表示部にDEGを表示させます 4[*][cos]25 [↵]	4*COS25_ 3.625231148	3.625231148
2	$\cos\frac{\pi}{5}$	表示部にRADを表示させます [cos][(][π][/]5[)] [↵]	COS(PI/5)_ 0.809016994	0.809016994
3	$\cos(473^g+168^g)$	表示部にGRADを表示させます [cos][(]473[+]168[)] [↵]	COS(473+168)_ −0.799684658	−0.799684658

③ TAN関数

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例	題	キー操作	表示部	答
1	$\tan 19.7^\circ$	表示部にDEGを表示させます [tan]19.7 [↵]	TAN19.7_ 0.358051837	0.358051837
2	$\tan(24^\circ 8' 55'' + 37^\circ 19' 23'')$	表示部にDEGを表示させます [tan][(][→DEG]24.0855[+] [→DEG]37.1923[)] [↵]	TAN(DEG24.0855+ DEG37.1923)_ 1.839600915	1.839600915
3	$\tan(\pi+\frac{3}{8}\pi)$	表示部にRADを表示させます [tan][(][π][+]3[/]8[*] [π][)] [↵]	TAN(PI+3/ 8*PI)_ 2.414213562	2.414213562

14 逆三角関数 [2nd F][sin⁻¹], [2nd F][cos⁻¹], [2nd F][tan⁻¹]

逆三角関数とは、三角関数　sin, cos, tanのそれぞれの逆関数です。

逆関数とは、$y=x$　という一次関数に対して対称になる関数をいいます。

常用対数 ←→ 常用指数
互いに逆関数の関係

自然対数 ←→ 自然指数
互いに逆関数の関係

これと同じように、

三角関数 ←→ 逆三角関数
互いに逆関数の関係

それぞれ$x=y$に対して対称な関数となります

$y=\sin x$ ←→ $x=\sin^{-1}y$　（アークサインワイと読む）

$y=\cos x$ ←→ $x=\cos^{-1}y$　（アークコサインワイと読む）

$y=\tan x$ ←→ $x=\tan^{-1}y$　（アークタンジェントワイと読む）

逆三角関数の計算は、三角関数と同様に　DEG、RAD、GRADの指定が必要です。

今、上の式のxをθとおけば　$(x=\theta)$

三角関数	逆三角関数	逆三角関数の値の範囲		
		DEG	RAD	GRAD
$y=\sin\theta$	$\theta=\sin^{-1}y$	$-90^\circ\leqq\theta\leqq 90^\circ$	$-\frac{\pi}{2}\leqq\theta\leqq\frac{\pi}{2}$	$-100^g\leqq\theta\leqq 100^g$
$y=\cos\theta$	$\theta=\cos^{-1}y$	$0^\circ\leqq\theta\leqq 180^\circ$	$0\leqq\theta\leqq\pi$	$0^g\leqq\theta\leqq 200^g$
$y=\tan\theta$	$\theta=\tan^{-1}y$	$-90^\circ\leqq\theta\leqq 90^\circ$	$-\frac{\pi}{2}\leqq\theta\leqq\frac{\pi}{2}$	$-100^g\leqq\theta\leqq 100^g$

●三角関数　$\sin\theta$、$\cos\theta$の値は、すべて－１から＋１の値にありますから、それ以外の値で$\sin^{-1}y$、$\cos^{-1}y$の計算はできません。

[1] $\sin^{-1}$, $\cos^{-1}$, $\tan^{-1}$

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

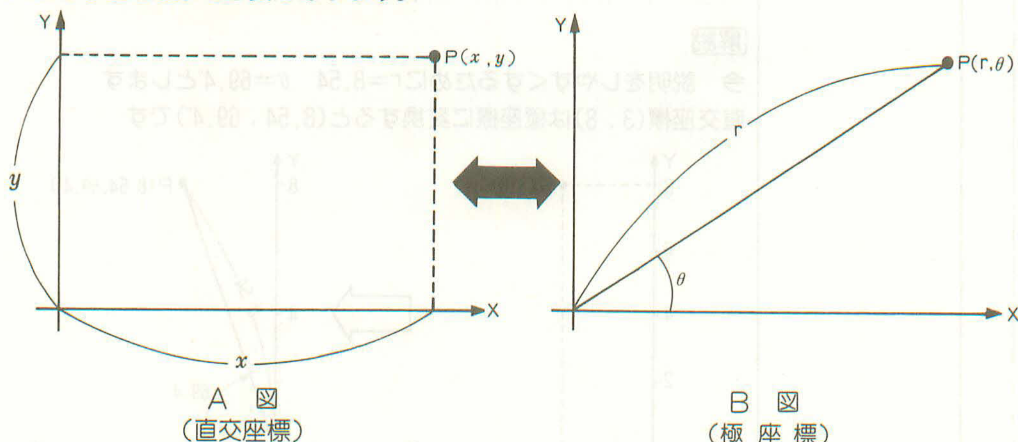
	例　題	キー操作	表示部	答
1	$\sin^{-1}0.5$(DEG)	表示部にDEGを表示させます [2ndF][$\sin^{-1}$]0.5 [↵] または 0.5[2ndF][$\sin^{-1}$]	ASN0.5_ 30. 30.	30 30
2	$\cos^{-1}0.628$ (RAD)	表示部にRADを表示させます [2ndF][$\cos^{-1}$]0.628 [↵] または 0.628[2ndF][$\cos^{-1}$]	ACS0.628_ 0.891815777 0.891815777	0.891815777 0.891815777
3	$\tan^{-1}1$ (GRAD)	表示部にGRADを表示させます [2ndF][$\tan^{-1}$]1 [↵] または 1[2ndF][$\tan^{-1}$]	ATN1_ 50. 50.	50 50
4	$2\sin^{-1}0.785$ (DEG)	表示部にDEGを表示させます 2[*][2ndF][$\sin^{-1}$]0.785 [↵]	2*ASN0.785_ 103.4413565	103.4413565
5	$\cos^{-1}0.43+$ $\cos^{-1}0.66$ (RAD)	表示部にRADを表示させます [2ndF][$\cos^{-1}$].43[+][2ndF][$\cos^{-1}$] .66 [↵]	ACS.43+ACS.66_ 1.976281116	1.976281116
6	$\tan^{-1}\sqrt{\frac{1-0.6^2}{0.6}}$ (DEG)	表示部にDEGを表示させます [2ndF][$\tan^{-1}$][√][(][(] 1[−].6[y^x]2[)][/].6[)] [↵]	ATN SQR((1−.6 ∧2)/.6)_ 45.92428582	45.92428582

15 座標変換

座標変換とは、ベクトルや複素数においてx成分、y成分で表される式を、大きさr、偏角θに変換することや、その逆に、大きさr、偏角θから、x成分、y成分に変換することをいいます。

x成分、y成分として表されるものを、**直交座標** $z=x+y\ i$
大きさr，偏角θとして表されるものを **極座標** $z=(r, \theta)$
と、いいます。

この座標変換の場合も、三角関数同様、角度を扱いますので、角度指定DEG、RAD、GARDのいずれかを指定する必要があります。



A 図
（直交座標）

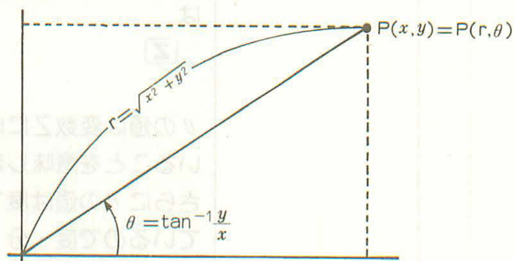
B 図
（極 座 標）

A図も、B図もXY平面上の点Pを表現する方法です。表現方法は違いますが、点Pの位置を示していることにおいては同じ結果となります。
XY平面上において、点Pの位置を表す方法として

A図のような表し方を ⟶ **直 交 座 標** といいます。
また
B図のような表し方を ⟶ **極 座 標** といいます。

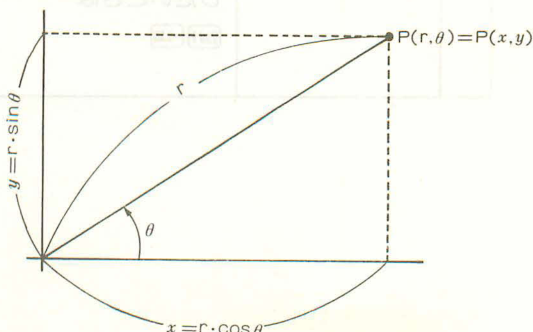
①直交座標を極座標に変換するには

● 極座標のrを求める ⟶ $r=\sqrt{x^2+y^2}$
● 偏角θを求める ⟶ $\theta=\tan^{-1}\frac{y}{x}$



②極座標を直交座標に変換するには

● 直交座標のxを求める ⟶ $x=r\cos\theta$
● 直交座標のyを求める ⟶ $y=r\sin\theta$



1 直交座標→極座標変換 [2nd F][→rθ]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例	題	キー操作	表示部	答
1	(3,8)を極座標に変換 (P=3+8i)	角度単位を"度"に指定します 表示部にDEGを表示させます [2nd F][→rθ][(]3[,]8[)] [↵]	POL(3,8)_ 8.544003745	rの値↓ 8.544003745
		[Z] [↵]	69.44395478	69.44395478 ↑ θの値
		[解説] 今　説明をしやすくするためにr=8.54　θ=69.4°とします 直交座標(3，8)は極座標に変換すると(8.54，69.4°)です		
		ここでｒの値を確認したいときは [Y] [↵] これはｒの値が変数Ｙに収められていることを意味します	Y_ 8.544003745	8.544003745
		次にθの値を確認したいときは [Z] [↵] θの値は変数Ｚに収められていることを意味します	Z_ 69.44395478	69.44395478
		さらにθの値は度で表わされているので度・分・秒に変換したいときは [2nd F][→D.MS]	69.26382372	69度26分38.2372秒

	例　題	キー操作	表示部	答
2	(3, −8) を極座標に変換 (P=3−8 i)	表示部にDEGを表示させます 2ndF →rθ (3 , − 8) ↵ Z ↵	POL(3,−8)_ 8.544003745 Z_ −69.44395478	rの値 8.544003745 −69.44395478 θの値
		解説 r=8.54、θ=−69.4°とすると 直交座標(3,−8)は極座標に変換すると(8.54,−69.4°)です		
3	(−3, −8) を極座標に変換 (P=−3−8 i)	表示部にDEGを表示させます 2ndF →rθ (− 3 , − 8) ↵ Z ↵	POL(−3,−8)_ 8.544003745 Z_ −110.5560452	8.544003745 −110.5560452
		解説 r=8.54、θ=−110.5°とすると 直交座標(−3, −8)は極座標に変換すると (8.54,−110.5°)です		

② 極座標 → 直交座標変換 [2nd F][→xy]

例題をはじめる前に[CLS]キーを押して表示をクリアしてください。

例題		キー操作	表示部	答
1	(2,30°)を直交座標に変換 P=(2,30°)	表示部にDEGを表示させます [2nd F][→xy][(]2[,]30[)] [↵] [Z] [↵]	REC(2,30)_ 1.732050808 Z_ 1.	xの値 ↓ 1.732050808 yの値 → 1
		●xの値を再確認するときは [Y] [↵] 1.732050808 xの値は変数Yに収められていることを意味します ●yの値を再確認するときは [Z] [↵] 1. yの値は変数Zに収められていることを意味します		
		解説 極座標(2,30°)は、直交座標(1.73,1)です		
2	(2,−30°)を直交座標に変換	表示部にDEGを表示させます [2nd F][→xy][(]2[,][−]30[)] [↵] [Z] [↵]	REC(2,−30)_ 1.732050808 Z_ −1	1.732050808 −1
3	$(2,\frac{\pi}{3})$ を直交座標に変換 $P=(2,\frac{\pi}{3})$	表示部にRADを表示させます [2nd F][→xy][(]2[,][π][/]3[)] [↵] [Z] [↵]	REC(2,PI／3)_ 1. Z_ 1.732050808	1 1.732050808

16 統計計算

たくさんのデータをひとつのまとまった資料にするためには、統計的な処理が必要です。

本機は統計モードで、一変数統計計算、二変数統計計算を行うことができます。

一変数統計計算は、たとえばテストの点数のような1種類のデータを用いて、その平均値、標準偏差などの統計量を求めることができます。

二変数統計計算は、たとえば身長と体重のように関連がある(と予想される)2種類のデータを用いて、それぞれのデータの平均や標準偏差を求めたり、2種類のデータの相関関係を調べたり、また一次回帰線による推定を行うことができます。

① 統計モードの設定と解除の方法

1．統計モードにするときは

RUN（またはPRO）モードで

2ndF STAT

と押します。統計モードになり、右の一変数／二変数選択画面になります。

なお、統計モードでは、画面右下に"STAT"シンボルが点灯します。

```
*****  トウケイ  ブ゛ンセキ  *****
   1：1ヘンスウ  トウケイ（x）
   2：2ヘンスウ  トウケイ（x，y）
ハ゛ンコ゛ウ  ヲ  エランテ゛  クタ゛サイ.
                                STAT
```

この選択画面のとき、1キーで一変数統計計算、2キーで二変数統計計算を選ぶことができます。

2．統計モードを解除するときは

2ndF STAT

と押します。RUNモードになります。

② 一変数統計計算

1．一変数統計計算で求める統計量

一変数統計計算では、次の統計量を求めることができます。

・サンプル数　n

・サンプルの総和　Σx

・サンプルの2乗の和　Σx^2

・平均　$\bar{x}=\frac{\Sigma x}{n}$

・サンプルの標準偏差　$s=\sqrt{\frac{\Sigma x^2-n\bar{x}^2}{n-1}}$

母集団より抽出されたサンプルデータから、母集団の標準偏差を推定する場合に使用します。

・母標準偏差　$\sigma=\sqrt{\frac{\Sigma x^2-n\bar{x}^2}{n}}$

母集団のすべてをサンプルデータとして、その標準偏差を求める場合、またはサンプルを母集団とみなして、その標準偏差を求める場合に使用します。

2. 一変数統計計算の選択

2ndF STAT と押して統計モードにしたあと、画面に従って 1 キーを押せば一変数統計計算が選択され、処理選択画面になります。

2ndF STAT

```
***** トウケイ ブンセキ *****
   1:1ヘンスウ トウケイ(x)
   2:2ヘンスウ トウケイ(x,y)
バンゴウ ヲ エランデ クダサイ.
```

1

```
*** ショリ ***      (x)
1:ニュウリョク     2:サクジョ/クリア
3:ブンセキ         4:プリンタ
バンゴウ ヲ エランデ  クダサイ.
```

処理選択画面では、それぞれ次の処理を選ぶことができます。

1 ……入力	データの入力を行うときに選びます。
2 ……削除/クリア	まちがったデータを入力した場合などに、そのデータを削除するときや、新たに統計計算を始める場合に、前に計算した統計量をすべて消去するときに選びます。
3 ……分析	統計量を求めるときに選びます。
4 ……プリンタ	統計量を印字するときに選びます。ただし、別売のプリンタが接続されているときにのみ選ぶことができます。プリンタが接続されていないときは 4 キーを押しても何も変わりません。

なお、一つ前の選択画面に戻すときは ON(BREAK) キーを押してください。

3. データの入力方法

処理選択画面で 1 キーを押せばデータ入力画面になり、データが入力できます。

1

```
  ** データ ニュウリョク **
1:x=_
```

└ データが入ります。

└ 何個目のデータかを示します。

①データを1つずつ入力する場合は

データ ↵

と押します。

②負のデータを入力する場合は

− データ ↵

と押します。

③同じデータが複数個ある場合は

データ [,] 個数 [↵]

と押します。

データをすべて入力したら [BREAK ON] キーを押します。処理選択画面に戻ります。

【補足】

統計では、同じデータが何個かある場合は"度数"と言う言葉を使います。

たとえば、同じデータが3個ある場合は"度数3"のように表します。

4．統計量を求める方法

処理選択画面で [3] キーを押せば、次の分析画面になります。

[3]

```
** ブンセキ **        (x)
1：n    2：Σx    3：Σx²   4：x̄
5：s    6：σ
バンゴウ ヲ エランデ クダサイ.
```

[1]～[6] キーで次の統計量を求めることができます。

[1] ……n　サンプル数

[2] ……Σx　サンプルの総和

[3] ……Σx²　サンプルの2乗の和

[4] ……x̄　平均値

[5] ……s　サンプルの標準偏差

[6] ……σ　母標準偏差

処理選択画面に戻すときは、[BREAK ON] キーを押します。

5．新たに統計計算を行うときは（前の計算内容を消すときは）

次の2つの方法があります。

①いったん統計モードを解除し、改めて統計モードにすれば消去されます。（統計モードを解除したとき、統計量の一部は変数に保存されています。(くわしくは72ページを参照)

②削除／クリア機能で消去します。まず、処理選択画面にして次のようにキーを押します。

[2]

削除／クリア選択画面になります。

```
** サクジョ/クリア **
 1：データ サクジョ
 2：オール クリア
バンゴウ ヲ エランデ クダサイ.
```

[2]
消去確認画面になります。

```
＊ オール クリア ＊
1：YES
2：NO
バ゛ンコ゛ウ ヲ エランテ゛ クダ゛サイ.
```

[1]　前の計算内容が消去され、処理選択画面に戻ります。

[2]を押した場合は、計算内容を消去せずに処理選択画面に戻ります。

■例　題

ある試験での点数を、ランダムに選び出した35人について見た場合、右のようになりました。
これより平均値、標準偏差を求めなさい。

No.	点数	人数	No.	点数	人数
1	30	1	5	70	8
2	40	1	6	80	9
3	50	4	7	90	5
4	60	5	8	100	2

キー操作	表示部
・統計モードにします。 [2ndF] [STAT]	
・一変数統計計算を選びます。 [1]	
・"入力"を選びます。 [1]	
・データを入力します。	
30 [↵] 40 [↵] 50 [,] 4 [↵]	
60 [,] 5 [↵] 70 [,] 8 [↵]	
80 [,] 9 [↵] 90 [,] 5 [↵]	
100 [,] 2 [↵]	34：x=100.,2.
これでデータの入力は終わりです。	36：x=_
・処理選択画面に戻します。 [BREAK ON]	
・"分析"を選びます。 [3]	
・平均値を求めます。 [4]	$\bar{x}$= 71.42857143
・サンプルの標準偏差を求めます。 [5]	s= 16.47508942
・母標準偏差を求めます。 [6]	σ= 16.23802542
・処理選択画面に戻します。 [BREAK ON]	

＊＊サンプル数、総和、２乗の和を求めるときは、分析画面でそれぞれ[1]、[2]、[3]キーを押します。

途中結果として平均値や標準偏差などの統計量を求めた後、もう一度、処理選択画面で"入力"を選んでデータを入力すれば、続きのデータとして入力できます。

6．データの削除

データ入力でまちがったデータを入れた場合などに使用する機能です。

処理選択画面にして次のようにキーを押すとデータ削除画面になります。

[2]

```
** サクジ゛ョ／クリア **
 1：デ゛ータ サクジ゛ョ
 2：オール クリア
バ゛ンコ゛ウ ヲ エランデ゛ クダ゛サイ.
```

[1]

```
* デ゛ータ サクジ゛ョ *
x＝_
```

この画面で、データ入力のときと同様の操作で、まちがったデータ、削除したいデータを入力すれば削除が行われます。データ入力のときと同様に[,]を使って複数個のデータを削除することもできます。

削除を行ったあと、データ入力画面にして正しいデータを入力すれば、データの訂正ができます。

7．統計量の印字

データを入力した後、計算された統計量を別売のプリンタCE-126Pで印字することができます。

プリンタを本機に接続して電源を入れた後、データを入力して、処理選択画面で“プリンタ”を[4]キーで選べば印字が開始されます。

```
*** ショリ ***          (x)
1：ニュウリョク      2：サクジ゛ョ／クリア
3：フ゛ンセキ        4：フ゜リンタ
バ゛ンコ゛ウ ヲ エランデ゛ クダ゛サイ.
```

[4]

```
** インジ゛チュウ **
```

印字が終われば処理選択画面に戻ります。

〈印字例〉68ページの例題のデータが入っている場合

```
n=                 35.
Σx=              2500.
Σx²=           187800.
MEAN(x)=   71.42857143
s=         16.47508942
σ=         16.23802542
```

③ 二変数統計計算

二変数統計計算の基本的な操作方法は、一変数統計計算と同じですので、先に一変数統計計算の説明をお読みください。

1．二変数統計計算で求める統計量

二変数統計計算では、次の統計量を求めることができます。

・n、Σx、Σx²、$\bar{x}$は一変数統計計算と同じ。sx、σxは一変数統計計算のs、σと同じ。

・Σy		サンプル(y)の総和
・Σy²		サンプル(y)の2乗の和
・Σxy		サンプル(x，y)の積の和
・$\bar{y}$	$\bar{y}=\frac{\Sigma y}{n}$	サンプル(y)の平均値
・sy	$sy=\sqrt{\frac{\Sigma y^2-n\bar{y}^2}{n-1}}$	サンプル(y)から求める、母数を(n−1)としたときの標準偏差
・σy	$\sigma y=\sqrt{\frac{\Sigma y^2-n\bar{y}^2}{n}}$	サンプル(y)から求める、母数を(n)としたときの標準偏差
・a	$a=\bar{y}-b\bar{x}$	一次回帰線 y＝a＋bx の係数
・b	$b=\frac{Sxy}{Sxx}$	一次回帰線 y＝a＋bx の係数
・r	$r=\frac{Sxy}{\sqrt{Sxx\cdot Syy}}$	相関係数
・x′	$x'=\frac{y-a}{b}$	推定値（yの値からxの値を推定する）
・y′	$y'=a+bx$	推定値（xの値からyの値を推定する）

【補足】

$$Sxx=\Sigma x^2-\frac{(\Sigma x)^2}{n}$$

$$Syy=\Sigma y^2-\frac{(\Sigma y)^2}{n}$$

$$Sxy=\Sigma xy-\frac{\Sigma x\cdot\Sigma y}{n}$$

2．二変数統計計算の選択

[2ndF] [STAT] と押して統計モードにした後、画面に従って [2] キーを押せば二変数統計計算が選択されて、処理選択画面になります。

3．データの入力方法

処理選択画面で[1]キーを押せばデータ入力画面になります。画面に従って、x，yのデータを入力してください。

①データが1個（1組）の場合

データx [↵] データy [↵]

と押します。

②同じデータが複数個ある場合

データx [↵] データy [,] 個数 [↵]

と押します。

③負数のデータは、それぞれのデータの前に [−] キーを押します。

データをすべて入力したら [ON](BREAK) キーを押します。処理選択画面に戻ります。

4．統計量を求める方法

処理選択画面で [3] キーを押せば、分析画面になります。二変数統計計算では分析画面が2画面あります。画面の切り替えは [↓] [↑] キーで行います。

処理選択画面で[3]キーを押した場合

（"分析" の第1画面です）

```
** ブンセキ **      (x, y)      ↓
1:n     2:Σx     3:Σx²   4:x̄
5:sx    6:σx     7:Σy    8:Σy²
バンゴウ ヲ エランデ クダサイ.
```

[↓]

（"分析" の第2画面です）

```
** ブンセキ **      (x, y)      ↑
1:Σxy   2:ȳ      3:sy    4:σy
5:a  6:b  7:r   8:x'    9:y'
バンゴウ ヲ エランデ クダサイ.
```

[↑] キーで前の（第1）画面に戻ります。

■例　題

次の表は、ある地方の山桜の開花日（4月）と同地3月の平均気温の表です。

これより、一次回帰線 y＝a＋bx の係数a，bと相関係数rを求め、3月の平均気温が9.1度の場合の開花日および4月10日に開花した年の3月の平均気温を推定します。

年	1	2	3	4	5	6	7	8
平均気温(x度)	6.2	7.0	6.8	8.7	7.9	6.5	6.1	8.2
開 花 日(y日)	13	9	11	5	7	12	15	7

キー操作	表示部
・統計モードにします。 2ndF STAT	
・二変数統計計算を選びます。 2	
・"入力"を選びます。 1	
・データを入力します。	
6.2 ↵ 13 ↵ 7.0 ↵ 9 ↵	
6.8 ↵ 11 ↵ 8.7 ↵ 5 ↵	
7.9 ↵ 7 ↵ 6.5 ↵ 12 ↵	
6.1 ↵ 15 ↵ 8.2 ↵ 7 ↵	8：x=8.2
	y=7.
これでデータの入力は終わりです。	9：x=_
・処理選択画面に戻します。 BREAK ON	
・"分析"を選びます。 3	
・第2画面を呼び出します。 ↓	
・係数aを求めます。 5	a=　34.44951017
・係数bを求めます。 6	b=　−3.425018839
・相関係数rを求めます。 7	r=　−9.691068372E−01 (−0.969106837)
・開花日を推定します。 9	x=_
平均気温を入力　9.1 ↵	y=　3.281838734
	(推定：4月3日ごろ開花)
・"分析"の第2画面に戻します。 BREAK ON	
・平均気温を推定します。 8	y=_
開花日を入力　10 ↵	x=　7.13850385
	(推定：3月の平均気温は約7.1℃)
統計モードを解除　2ndF STAT	

【補足】

統計計算の統計量のうち、次のものは変数U～Zに入れられ、統計モードを解除しても保持されています。したがって、RUNモードでも、この統計量を使って計算ができます。

変数		U	V	W	X	Y	Z
統計量	一変数統計	—	—	—	Σx^2	Σx	n
	二変数統計	Σy^2	Σy	Σxy	Σx^2	Σx	n

なお、この内容は統計モードになったときに消去されます。

第3章　算術代入計算

1.算術代入計算

今までは、数値と数値の計算のやり方などを練習してきましたが、複雑な数式や数値のたくさんあるような式を計算処理していくときに、どうしても数式（数値を代入するので）が長くなって、しまいます。

こんなとき、あらかじめ数値を変数キー（アルファベットキー）に記憶させておき、文字と文字の計算によって答えを求めることができます。

①変数として使える文字

変数として使える文字は、[英文字]、[英文字][英文字]の組み合わせ、[英文字][数字]の組み合わせの3種類で、最高2文字分の長さまで使えます。

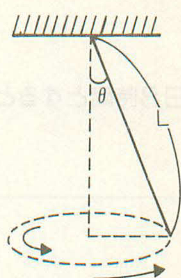
種　類	例	
1．[英文字]（アルファベット）	[A]～[Z]	A=1 B=100 Z=2.5
2．[英文字] [英文字]	[A]～[Z]	AA=1.5 MM=30.5 LX=0.003
3．[英文字] [数　字]	[A]～[Z] [0]～[9]	A1=6.28 G1=9.8 T2=358.2 X1=3 Y2=6

注1）アルファベットの小文字は大文字と同じ扱いになります。変数としてアルファベットの小文字を使用しても、それは大文字に変換されます。

注2）変数U、V、W、X、Y、Zは座標変換や統計計算で使用するため、変数として使うときには注意が必要です。

2. 例題と解説

■例　題①



左図のような円すい振り子の周期Tは、糸の長さをL、糸の鉛直となす角をθとすると、

$$T=2\pi\sqrt{\frac{L\cos\theta}{g}}\qquad g=9.8〔m／sec^2〕$$

で求めることができます。L=50cm　θ=25°の場合の周期Tを求めなさい。

このような例題をプログラム化しますと、127ページの例題㉘のようなものが考えられます。

■解　説①

計算の内容	キ　ー　操　作	表　示　部
最初変数の内容をすべて消去しておきます	表示部にDEGを表示させます。 [C][L][E][A][R] [↵]	CLEAR_ >
g=9.8 Gに9.8を代入します	[G][2ndF][=]9.8 [↵]	9.8
L=0.5 Lに0.5を代入します	[L][2ndF][=]0.5 [↵]	0.5
θ=25° θがありませんのでSに25を代入します	[S][2ndF][=]25 [↵]	25.
$T=2\pi\sqrt{\frac{L\cos\theta}{g}}$ 文字式として入力します	[T][2ndF][=]2[*][π][*][√][(][L][*][cos][S][/][G][)] [↵]	T=2*PI*SQR(L*COS S/G)_ 1.351106827 答　1.35秒
ここで入力した式を確認したい場合（プレイバックします。）	[►] 式の先頭が呼び出されます [↵]キーを押すと再び計算が行われます [↵] [◄] 式の後部が呼び出されます [↵]キーを押すと再び計算が行われます [↵]	T=2*PI*SQR(L*COS S/G) 1.351106827 T=2*PI*SQR(L*COS S/G)_ 1.351106827
	計算の結果はTに入っています [T] [↵] と押せば呼び出されます	1.351106827

文字式を入力した後で[↵]キーを押し、表示部にエラーが出たときは[◄]または[►]を押すと、エラーになった位置にカーソルが表示されます。訂正しなおして再び[↵]を押すと答が得られます。

■例　題②

月の軌道を地球を中心とする円とみなし、地球の半径ｒ、月の周期をＴとすると月の軌道半径は、

$$R=\sqrt[3]{\frac{g\cdot r^2\cdot T^2}{4\pi^2}}$$

で与えられます。 地球の半径ｒ＝6.4×10^6〔m〕、月の周期Ｔ＝27日８時間とすると Ｔ＝2.3616×10^6〔sec〕です。月の軌道半径を求めなさい。

■解　説②

計算の内容	キ　ー　操　作	表　示　部
	CLEAR ↵	
R＝6.4×10^6	R 2ndF ＝ 6.4 Exp 6 ↵	R＝6.4E6_ 6400000.
T＝2.3616×10^6	T 2ndF ＝ 2.3616 Exp 6 ↵	T＝2.3616E6_ 2361600.
G＝9.8	G 2ndF ＝ 9.8 ↵	9.8
$R=\sqrt[3]{\frac{g\cdot R^2\cdot T^2}{4\pi^2}}$ ここでRは地球の半径で使っているので月の軌道半径の変数をＲＲとします	R R 2ndF ＝ 2ndF ∛ (G * (R y^x 2) * (T y^x 2) / (4 * (π y^x 2))) ↵	RR=CUR(G*(R∧2)*(T∧2)/(4*(PI∧2)))_ 384190235.2〔m〕 答384190km

このように、原式にもとづいてあらかじめ原式の文字に対する数値を記憶させ、その後、文字式の演算で行う方法をとると、あとで式の確認も容易なので、とても便利な使い方のひとつです。

■例　題③

左図のようなＲＬ直列回路に流れる電流は、スイッチSWを閉じてからt秒後に

$$i(t)=\frac{E}{R}(1-e^{-\frac{R}{L}t})$$

となります。

今、Ｅ＝4.5〔V〕 Ｌ＝160〔mH〕 Ｒ＝55〔Ω〕として t=0.1〔ms〕における電流の値を求めなさい。

このような例題をプログラム化しますと、128ページの例題㉙のようなものが考えられます。

■解　説③

計算の内容	キー操作	表示部
	C L E A R ↵	
E=4.5	E 2ndF = 4.5 ↵	4.5
L=160〔mH〕		
$=160\times10^{-3}$	L = 160 Exp − 3 ↵	0.16
R=55	R 2ndF = 55 ↵	55.
T=0.1〔ms〕		
$=0.1\times10^{-3}$	T 2ndF = 0.1 Exp − 3 ↵	0.0001
$I=\frac{E}{R}(1-e^{-\frac{R}{L}t})$	I 2ndF = E / R * (1 −	I=E/R*(1−
	2ndF e^x (− R * T / L)	EXP (−R*T/L))_
	) ↵	0.002764709
		答 2.7647〔mA〕

■例　題④

物質の崩壊時間は　$t=\frac{1}{\lambda}\ln(1+\frac{Dt}{Pt})$　で求められます。

Dt=t時間後の同位原子数　　$Dt=5\times10^4$個
Pt=t時間後の安定原子数　　$Pt=3\times10^6$個
λ=崩壊定数　　λ=20.5

このとき、この物質は何時間経過しているでしょうか。

■解　説④

計算の内容	キー操作	表示部
	C L E A R ↵	
$Dt=5\times10^4$	D T 2ndF = 5 Exp 4 ↵	50000.
$Pt=3\times10^6$	P T 2ndF = 3 Exp 6 ↵	3000000.
λ=20.5	V 2ndF = 20.5 ↵	20.5
$t=\frac{1}{\lambda}\ln(1+\frac{Dt}{Pt})$	T 2ndF = 1/X V * ln (1 +	T=RCP V*LN (1+
	D T / P T)	DT/PT)_
	↵	0.000806307
		答 0.000806307時間

第4章　BASIC言語

㊟本章のプログラム例や解説で使用している数字とアルファベットは、以下の文字です。参考にしてください。

0123456789

ABCDEFGHIJKLMNOPQRSTUVWXYZ

1．BASIC言語をマスターする第一歩

①BASICとは

B；Beginner's　（ビギナーズ）……………　初心者向きで
A；All-Purpose　（オールパーパス）………　あらゆる目的に合う
S；Symbolic　（シンボリック）…………　記号を使った
I；Instruction　（インストラクション）…　命令
C；Code　（コード）……………………　語

それぞれの頭文字をとって、BASIC(ベーシック)という名のついたコンピュータ言語です。
BASICはやさしい英語と記号でできており、コンピュータがどんな仕事をすればよいのかを示す一連の命令文、すなわち「プログラム」を人間とコンピュータが対話するような形で作っていくことができるのが大きな特長です。

②プログラムとは

プログラムとは、計算機（コンピュータ）に計算などをおこなうための手順を指令する命令書のようなものです。この指令をコンピュータが理解できるように書き表したり、コンピュータに記憶をさせたりすることをプログラミングとか、プログラムを組むと言います。しかし、コンピュータでいろいろな問題を処理しようとするとき、最初からＢＡＳＩＣ言語でプログラミングする方法もありますが、ここではＢＡＳＩＣ言語のプログラムをマスターする第一歩として、問題を処理する方法の考え方や、手順を整理し、図的に順序だてて、プログラムの流れをわかりやすく表現することからはじめます。

③PC-E200の主な記号の読み方と意味

記号	読み方	意味
＋	プラス	加算
－	マイナス	減算
*	アスタリスク	乗算
／	スラッシュ	除算
＝	イコール	等号
＞	グレーター・ザン	より大きい
＜	レス・ザン	より小さい

記号	読み方	意味
"	ダブルクォーテーション	引用符
.	ピリオド	小数点
,	コンマ	区切
:	コロン	命令文を続けて書く
;	セミコロン	表示部をつめる
∧	べき（ハットマーク）	べき乗記号
#	クロスハッチ	番号記号

記号	読み方
!	イクスクラメーションマーク
?	クエッションマーク
@	アットマーク
&	アンドマーク又はアンパサイド
%	パーセントマーク
$	ドルマーク
'	シングルクォーテーション

記号	読み方
[ ]	大カッコ
{ }	中カッコ
¥	円記号
¦	パイプ記号
～	波形
_	アンダーライン

④「流れ図」（フローチャート，Flow chart）について

プログラムの処理方法の考え方や手順を整理し、図的に順序だてて書いたものを「流れ図」またはフローチャートといいます。
流れ図は、図記号と簡単な式や文字であらわします。図記号は、それぞれひとつひとつ意味を持っています。次に代表的な図記号とその意味、そして流れ図の基本形を示しますので十分に理解していただきたいと思います。

⑤代表的な「流れ図」とその意味

流れ図の記号	意味
1.	**端　子**　(Terminal Interrupt) 流れ図の開始，終了などの端子を表します。
2.	**処　理**　(Process) わく内（わく外に書いてもよい）に書かれてある処理を行います。
3.	**準　備**　(Preparation) 初期値などの準備などに用います。
4.	**入出力**　(Input／Output) 情報の入出力を意味します。入出力一般として用いられます。 また、本書では図記号 ▱ をRESTOREとみなします。
5.	**手操作入力**　(Manual Input) 変数への入力など、キーボードなどから手で操作して入力することを表します。
6.	**定義済み処理**　(Predefined Process) サブプログラムなど、別の場所で定義されている命令群などの処理を表します。
7.	**表　示**　(Display) 情報を人間が利用できるように、ディスプレイに表示します。
8.	**判　断**　(Decision) 判断・比較を行います。
9.	**書　類**　(Document) 書類を媒体とする入出力機能を表します。
10. FOR I=1～10,2 ≈ I	**くり返しループ** FOR～NEXT文に用いる記号 FOR IのIと ◇I が対応します。

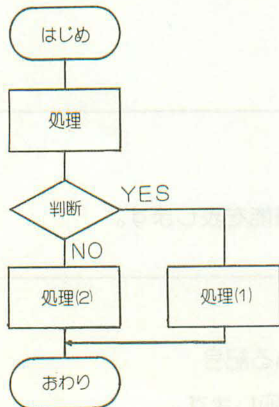
11.	**結 合 子**　(Connecter) 流れ図のほかの場所への出口、または、ほかの場所からの入口を表します。
12.	**紙カード**　(Punched card) 紙カードを媒体とする入出力機能を表します。
13.	**カセットテープ** カセットテープでのデータ入出力を表します。
14.	**流 れ 線**　(Flow Line) 記号を結びつける機能を表します。
15.	**注　釈**　(Comment Annotation) 明りようにするために、説明または注意を加える機能を表します。

⑥プログラムの形（構造）

1．直線形の流れ図（シーケンス構造）

- 直線形プログラム
- 「判断」が入っていません。

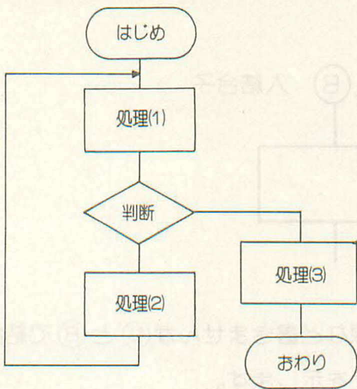
2．分岐形の流れ図（IF～THEN構造）



- 分岐形プログラム
- 設定された条件によってプログラムの流れが分岐します。
- もし、条件が満たされたら処理(1)を行い，
  IF　THEN
  そうでなければ、処理(2)を行います。

3．ループ形の流れ図（ループ構造）

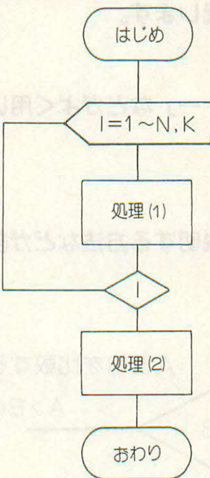
＜その１＞



●くり返し形プログラム〈その１〉

●条件が満たされるまでくり返し、処理(1)と処理(2)をします。条件が満たされると、このループを抜けて次の処理(3)へ向います。

＜その２＞



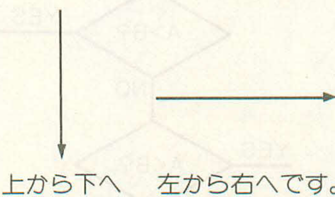
●くり返し形プログラム〈その２〉

●条件が満たされている間は、処理(1)をくり返します。

●条件が満たされなくなると、ループを抜けて次の処理(2)へ向います。

⑦「流れ図」全体の約束について

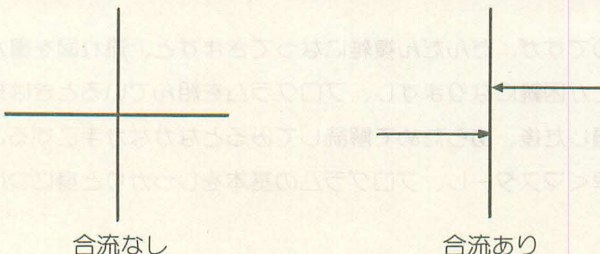
1．流れ図における流れ線の方向は、原則として、



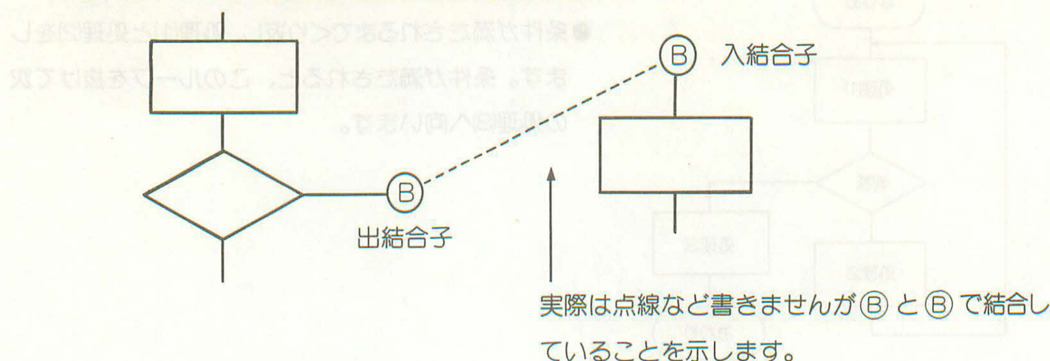
上から下へ　左から右へです。

流れの方向が、これに合わないとき、流れを示す矢印を用います。

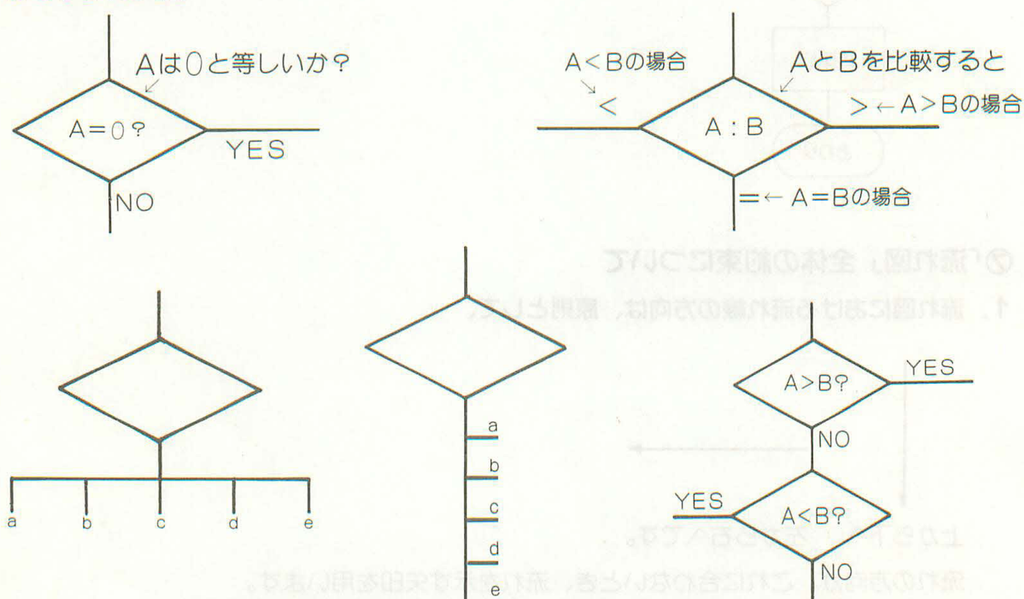
2．流れ線は、交差してもかまいません。交差しても論理的関係はありません。



3. 結合子は、流れ線が中断される点を表わすのに用いる出結合子と、中断された流れ線が再開される点を表すために用いる入結合子があります。それぞれの結合子に同じ文字や数字などを記入し、これらが結合していることを示します。



4. 1つの流れ図記号から出口を2つ以上書く場合は、必要な数だけの流れ線をその記号から出すか、あるいは、その記号から出た流れ線の数だけ分岐する形で表現します。
分岐の出口には、分岐条件を記入しておくことになっています。
「>, =, ≠, <」などや「YES」,「NO」あるいは「0, 1, +, −」などがよく用いられます。
比較記号としては「:」や「?」などが使われます。
また最近では、流れ図記号の中だけでなく、図記号の周辺に文章で説明する方法などが多くみうけられます。



プログラムが単純なうちはよいのですが、だんだん複雑になってきますと、流れ図を書かないとプログラムを組み立てることが困難になりますし、プログラムを組んでいるときは理解していても、しばらく時間が経過した後、あらためて解読してみるとなかなか手こずることがあります。流れ図の考え方を早くマスターし、プログラムの基本をしっかりと身につけましょう。

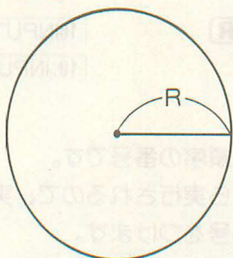
2. プログラムの基本

STEP ① INPUT,PRINT,END,GOTO文

〔例　題〕①

半径を入力して、円の面積を求める
プログラムを作りなさい。

〔解　説〕①



左図において、半径をR、面積をSとすると

$S=\pi R^2$

で面積を求めることができます。

流　れ　図（フローチャート）	文番号または行番号（ラインナンバー）	BASIC言語によるステートメント（ひとつの意味をもった処理式や、命令語のこと）
はじめ		BASIC を押して画面左下に“PRO”を点灯させます。
Rを入力する	10	INPUT　R
$S \leftarrow \pi R^2$	20	S=PI＊R∧2 S=PI＊R＊R でもよい
面積Sを表示	30	PRINT　S
おわり	40	END

（注）π はPIに変換されて入力されます。

①プログラムの入れ方

プログラムを計算機に書き込むときは、次の手順で行います。

操作の手順	操作手順の説明と表示内容
①. ONキーを押して電源を入れます。 BASICキーでプログラムモード（PROモード）にします。	プログラムを書き込むことのできる状態にします。
②. NEW↵	先に入力した（かもしれない）プログラムやデータをすべて消します。
③. 文番号を入れ、文を入れます。 10　INPUT R 文番号(行番号とは)→ (文番号は行番号(ラインナンバー)ともいいます。) 表示について→	1 0 SHIFT + Z(INPUT) R　[10INPUTR_] ↵　[10:INPUT R] プログラム文の実行順序の番号です。 数字の低い文番号から実行されるので、実行の順に数字の低い文番号をつけます。 文番号は、10番おきにつけるのが一般的です。後から、プログラムを追加するときに都合がよいからです。 なお、本機は1～65279までの整数を文番号として使用できます。 文番号に続けて文（ステートメント）を入れ、↵キーを押すと、文番号の後に自動的に：（コロン）が入ります。 10：INPUT␣R 自動的に1桁(1文字)あきます。 自動的に：(コロン) が入ります。
④. 20　S=PI*R∧2 (S=PI*R*Rでも同じです。)	文番号10で与えられたRの値でπ・R²を計算し、計算した数値をSという記号の中にしまっておきなさい（という意味です)。 2 0 S SHIFT + = π * R yˣ 2 ↵ [20:S=PI*R^2]
⑤. 30　PRINT　S	3 0 SHIFT + X(PRINT) S　[30PRINTS_] ↵　[30:PRINT S] 文番号20のSの記号の中に入っている数値を表示しなさい（という意味です)。
⑥. 40　END ENDの意味→	4 0 E N D ↵　[40:END] これでプログラムは終了しました（という意味です)。

[↵]キーについて	[↵]キーは文の終わりに必ず入れます。 これは「この文は終わりました」または「プログラム文として書き込むことが終了しました」という意味がふくまれています。

INPUTの入力方法———→	INPUTは[SHIFT]＋[Z](INPUT)と押してもよいですし [I][N][P][U][T] でも同じです。
PRINTの入力方法———→	PRINTは[SHIFT]＋[X](PRINT)と押してもよいですし [P][R][I][N][T]でも同じです。 このように、本機ではBASIC言語の命令語が、まだいくつかリザーブされていますので、活用してください。
ENDの入力方法	ENDは、[E][N][D]と押します。
文（ステートメント）について 〔例題〕①をマルチ・ステートメントにすると	1つの行（ライン）は1つ以上の文（ステートメント）からなり、2つ以上の文（マルチ・ステートメント）になる場合は文と文の間に：（コロン）を入れて区別します。（コロンを入れる） ＿＿ ＿＿：＿＿：＿＿ [↵] 文番号　文　文　文 10　INPUT　R：S=PI＊R∧2：PRINT S：END となります。ただし1行254文字以上は書けません。

②プログラムの修正・編集

プログラムを作成する場合、キー操作のミスなどにより正しいプログラムにならないことが多くあります。このような場合の処理は下表のキー操作で行います。

表示されている行（ライン）内の ① 訂　　正	◄または►キーでカーソルを移動し、 正しいキーの入力
② 削　　除 （Delete, デリート）	SHIFT＋INS（DEL）
（Backspace, バックスペース）	BS
③ 追　　加 （Insert, インサート）	INS

③プログラムの確認

プログラム全部の内容をコンピュータに書き込んだら、次にＰＲＯモードで下表のキー操作により、その内容の確認をします。

キー操作	説明
↓	このキーを1回押すと、現在表示されている次のラインを表示します。 このキーを押し続けると、順次、次のラインを表示します。 また、プロンプト記号(>)が表示されているときにこのキーを押すと、先頭のラインから、画面に表示できる範囲を表示します。 画面にカーソルが出ているときは、カーソルを画面の下の行または、プログラムラインの一番後へ移します。 すでにカーソルが一番後にあるときは、次のラインを表示します。
↑	このキーを1回押すと、現在表示されている前のラインを表示します。 このキーを押し続けると、順次、前のラインを表示します。 また、表示内容を一度消去してこのキーを押すと、最終のラインを表示します。 画面にカーソルが出ているときは、カーソルを画面の上の行または、プログラムラインの先頭へ移します。 すでにカーソルが先頭にあるときは、前のラインを表示します。
LIST↵ またはL.↵	先頭のラインから、画面に表示できる範囲を表示します。
LIST 30 ↵ または L.30 ↵	文番号（ラインナンバー）30行から、画面に表示できる範囲を表示します。(指定した文番号および、それより大きい文番号が存在しないときはエラー40になります)
プログラムを呼び出した後、►または◄キーを押せば表示部の1行目に表示されているプログラムラインにカーソルが現れます。►キーを押し続ければそのプログラムラインの最後までカーソルが行きます。◄キーの場合は先頭まで行きます。	

④文(ライン)の追加・変更・削除

今、確認したところ、下の表の右側のような誤りがあったとします。

正しいプログラム	誤って入力したプログラム
10 INPUT R 20 S=PI＊R∧2 30 PRINT S 40 END	10 INPUT R < ここに S=PI＊R∧2がない 20 PRINT S 30 END

誤って入力したプログラムの文番号10行と20行の間にS=PI＊R∧2を入れるために、新しく15という文番号を選びます。
15 S=PI＊R∧2 ↵ とすると、下表のように訂正されたプログラムができあがります。

```
10 INPUT R
15 S=PI*R∧2
20 PRINT S
30 END
```

プログラムの内容は同じです。文番号が異っていても問題はありません。

これで正しいプログラムが完成。

再び誤った入力プログラムにもどって文番号(ラインナンバー)も同じ（10, 20, 30, 40）にしたいときは、以下の手順で行います。

20 ↵ ……▶誤ったプログラムの20行はこの操作で削除されます。

20 S=PI＊R∧2 ↵ ……▶そこで新しく20行を入力します。

30 END ……▶つづいて、上の方法で30行も削除してもよいのですが、カーソルを左端に移動して（ ◄ キーを使う）文番号を訂正すると、プログラムは左のようになります。

4 ↵ ……ここにカーソルを移動します。

```
10 INPUT R
20 S=PI*R∧2
30 END
40 END
```

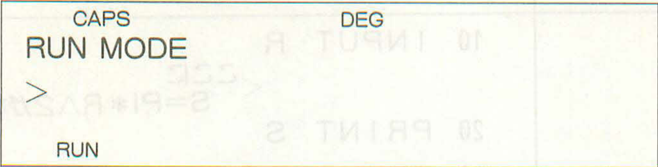
次にこの30行を削除(操作は 30↵)し、あらためて 30 PRINT S ↵ と入力すれば、文番号のそろったプログラムが完成します。

```
10 INPUT R
20 S=PI*R∧2
30 PRINT S
40 END
```

文番号の付け直しはRENUM命令(リナンバー)を使う方法もあります。「BASIC の各命令」の説明を参照してください。

⑤プログラムの実行

プログラムの実行は、実行モード（RUNモード）で行いますので BASIC キーを押して画面左下に"RUN" を表示させてください。



そして次の命令で実行を開始します。

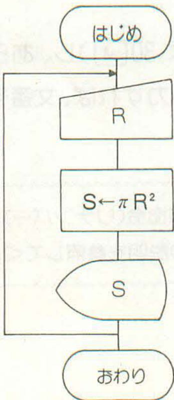
RUN ↵	最も低い数字の文番号より実行を開始します。
RUN 文番号 ↵	指定した文番号より実行を開始します。

それでは円の面積を求めるプログラムを実行させてみましょう。

実行の内容	キー操作	表示部
RUNモードに設定		>
実　行	SHIFT ＋ V(RUN) ↵ または R U N ↵	?
半径R＝1〔㎝〕 の場合	1 ↵	1_ 3. 141592654 >
半径R＝2〔㎝〕 の場合	SHIFT ＋ V(RUN) ↵ 2 ↵	? 2_ 12. 56637061 >
半径R＝3〔㎝〕、半径R＝4〔㎝〕の場合も計算してください。		

このようにしていくと、いちいち SHIFT ＋ V(RUN) ↵ とするのが、めんどうになってきます。そこで下の流れ図のように、Sの表示が終われば、再び先頭に戻る命令語があれば、いちいち RUN ↵ としなくともよくなります。

BASIC言語では、このようなときにGOTO文を用います。




```
10  INPUT  R
20  S=PI*R∧2
30  PRINT  S
35  GOTO  10
40  END
```

30行と40行の間に新しくGOTO文を入れるために

35　GOTO　10 ↵ と操作します。

このプログラムを実行すると、

実行の内容	キー操作	表示部
RUNモード	RUN ↵	?
R=1 次のRの入力待ち ——→	1 ↵	1_ 3.141592654 ?
R=2 次のRの入力待ち ——→	2 ↵	2_ 12.56637061 ?
このように、かんたんに計算をすすめることができます。		

このプログラムの実行を中止するときは BREAK ON キーを押してください。

⑥プログラムのアプリケーション

今までのプログラムの実行は入力・出力にただ数値が入っているだけで、ちょっと目をはなしたりすれば、それがどのような意味をもつ数値かわからなくなります。

ここではINPUT文とPRINT文に変化をもたせ、わかりやすい工夫を試みることにします。

```
10 INPUT R
```

※この部分をメッセージとして表示します。

```
10 INPUT "ハンケイ R=";R
```

セミコロン ［Rに入れる数値を入力したとき、その数値をメッセージに続けて表示させます。］

スペース ［つまり、空白。文字と記号を分離してわかりやすいようにするために1文字空白をつくりました。スペースは SPACE キーで入れます。］

※"ハンケイ R="のように" "引用符（ダブルクォーテーション）で囲まれたものは、数値と異なり文字として扱われます。

```
30 PRINT S
30 PRINT "R=";R;"  S=";S
```

これを入れないとRの数値とSの文字が、くっつきすぎてわかりにくくなるので␣␣スペースを2つ入れて、Rの数値とSの文字を離しておきます。

参考

" "で囲まれた中に入れたアルファベットの小文字は、そのまま小文字として扱われます。

" "で囲まれた中以外にアルファベットの小文字を用いると、大文字に変換されます。(ただし、注釈文は除く)

プログラムを整理すると下のようになります。

```
10  INPUT"ハンケイ␣R=";R
20  S=PI*R∧2
30  PRINT"R=";R;"␣␣S=";S
35  GOTO 10
40  END
```

これを実行してみましょう。

実行の内容	キー操作	表示部
RUNモード	RUN ↵	ハンケイ R=_
R=1 次のRの入力待ち——→	1 ↵	ハンケイ R=1_ R=1. S=3.141592654 ハンケイ R=_
R=2 次のRの入力待ち——→	2 ↵	ハンケイ R=2_ R=2. S=12.56637061 ハンケイ R=_

⑦エラー表示とその処理方法

プログラムを実行したときにプログラムに誤りがあったり、データが不適当な場合などでエラーが発生することがあります。

エラーが発生すると次のようなエラーメッセージが表示されます。

```
ERROR 10 IN 20
```

これは、文番号20行にエラーコード 10の内容のエラーが発生したことを意味します。

●エラーの処理手順

① CLSキーでエラーを解除します。
② BASICキーでPROモードにします。
③ ↓キーまたは↑キーを押します。
④ エラーの発生した文番号が表示されます。
⑤ エラーの発生した位置にカーソルが表示されています。
⑥ エラーの発生した原因を探して、プログラムの修正をします。

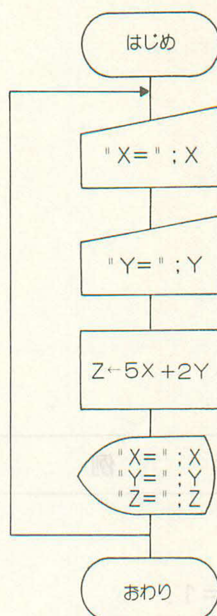
このコンピュータには多くのエラーコードがあります。

267ページにエラーコードの内容を示しますので、活用してください。

〔例　題〕②

ある適当な2つの数、XとYを入力して、5X+2Yを求めるプログラムをつくりなさい。

■フローチャート



入力前にBASICキーでPROモードにし、

NEW↵と操作します。

■プログラム例

```
10  INPUT"X=";X
20  INPUT"Y=";Y
30  Z=5*X+2*Y
40  PRINT"X=";X;"Y=";Y;"Z=";Z
50  GOTO 10
60  END
```

メモリ内容	
変　数	内　容
X	Xの値
Y	Yの値
Z	Z=5X+2Y

■数値代入例

Xの値	Yの値	Zの値(答)
1	2	9
5	−3	19
10	50	150
22	36	182

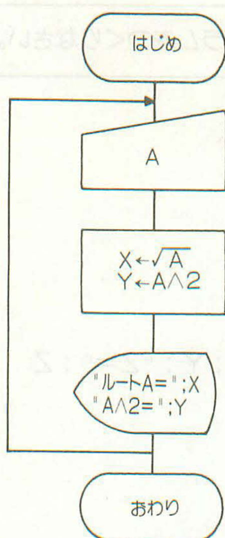
STEP② 切り捨て・四捨五入・桁指定

プログラムを実行し数値を求める場合、小数点以下2桁程度まで求めればよい場合が多くあります。このようなとき、切り捨て・四捨五入あるいは桁指定をして必要な桁だけ表示するプログラムについて次の例題で練習してみましょう。

〔例　題〕③

数Aを入力して$\sqrt{A}$, A^2の値を表示するプログラムをつくりなさい。
数Aは1、2、3、4、5.23を入力するものとします。

■フローチャート



■プログラム例

```
10  INPUT A
20  X=SQR A
30  Y=A∧2
40  PRINT"ルートA=";X;"␣A∧2=";Y
45  GOTO 10
50  END
```

■キー操作手順

操　　作			表　示　例
BASIC		RUNモードにします	
RUN	↵		?
1	↵	√1と1²の答 ——→	ルートA=1.　A∧2=1. ?
2	↵	√2と2²の答 ——→	ルートA=1.414213562 A∧2=4. ?
3	↵	√3と3²の答 ——→	ルートA=1.732050808 A∧2=9. ?
4	↵	√4と4²の答 ——→	ルートA=2.　A∧2=16. ?
5.23	↵	√5.23と5.23²の答 ——→	ルートA=2.286919325 A∧2=27.3529 ? 表示桁数が多くなり、A²の値が2行に別れてしまいました。

この〔例題〕③では、

```
40 PRINT"ルートA=";X;"␣A∧2=";Y
```

というように、セミコロンによって表示を継続させたので、1行に全部表示されなくなります。

しかし40行を次のように2つの行に分けることによって、それぞれの答えを1行に表示させることができます。

```
40 PRINT"ルートA=";X
43 PRINT"A∧2=";Y
```

しかし、小数点以下2桁程度を表示すればよい場合には、プログラムを次のように変更することによって$\sqrt{A}$、A^2を一度に表示できます。

①小数点第3位以下を切り捨てる場合　INT／整数化(Integer／インテジャー)

```
10 INPUT A
20 X=SQR A
25 X=INT(100*X)/100      ←
30 Y=A∧2
40 PRINT"ルートA=";X;"␣A∧2=";Y
45 GOTO 10
50 END
```

(Xを100倍にしたものを整数化し、それを100で割ることを意味しています。Yも同様な処理をすることができます。)

操　　　　作	表　　示　　部
BASIC　RUNモードにします	
RUN ↵	?
1 ↵	ルートA=1.　A∧2=1. ?
2 ↵	ルートA=1.41　A∧2=4. ?
3 ↵	ルートA=1.73　A∧2=9. ?
4 ↵	ルートA=2.　A∧2=16. ?
5.23 ↵	ルートA=2.28　A∧2=27.3529

②小数点第3位以下を四捨五入する場合

```
10 INPUT A
20 X=SQR A
25 X=INT(100*X+0.5)/100 ←
30 Y=A∧2
40 PRINT"ルートA=";X;"␣A∧2=";Y
45 GOTO 10
50 END
```

(Xを100倍したものに0.5を加え、それを整数化し100で割ることを意味します。Yも同様な処理をすることができます。)

操　　　　作	表　　示　　部
BASIC　RUNモードにします	
RUN ↵	?
1 ↵	ルートA=1.　A∧2=1. ?
2 ↵	ルートA=1.41　A∧2=4. ?
3 ↵	ルートA=1.73　A∧2=9. ?
4 ↵	ルートA=2.　A∧2=16. ?
5.23 ↵	ルートA=2.29　A∧2=27.3529 ↑ 小数点第3位を四捨五入した値

③数値の桁指定をする場合　USING命令

USING命令を用いて桁数を指定し、表示させます。〔例題〕③のXの値を小数点2桁まで指定するときは、

```
PRINT USING"##.##"
            ↑   └→小数部は2桁分
   符号として1桁必要
```

とします。ただし、小数点第3位以下は切り捨てとなります。
またYの値は符号の1桁と数値の2桁で、全部で3桁分の指定が必要です。

```
10 INPUT A
20 X=SQR A
30 Y=A∧2
40 PRINT USING"##.##";"ルートA=";X
43 PRINT USING"###";"A∧2=";Y
45 GOTO 10
50 END
```

操　　作	表　　示　　部
BASIC　RUNモードにします	
RUN ↵	?
1 ↵	ルートA= 1.00 A∧2= 1 ?
2 ↵	ルートA= 1.41 A∧2= 4 ?
3 ↵	ルートA= 1.73 A∧2= 9 ?
4 ↵	ルートA= 2.00 A∧2= 16 ?
5.23 ↵	ルートA= 2.28 A∧2= 27

STEP③関数を使うプログラム

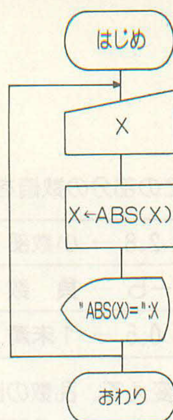
〔例　題〕④

数Xを入力してその絶対値を表示するプログラムをつくりなさい。

■解　説

ABS(X)の式でXの絶対値を求めます。〔Absolute(アブソリュート：絶対値)〕

■フローチャート



■プログラム例

```
10 INPUT X
20 X=ABS(X)
30 PRINT"ABS(X)=";X
40 GOTO 10
50 END
```

結果例	
X	ABS(X)
3	3
5	5
−2	2
−4	4
8	8

〔例　題〕⑤

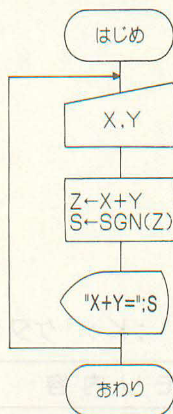
数XとYを入力して、X+Yの値がプラスなら1を、マイナスなら−1を、0なら0を表示するプログラムをつくりなさい。

■解　　説

符号はSGN(X)で調べることができます。〔Sign(シグナム)〕

SGN(X)で、X＞0のとき1、X=0のとき0、X＜1のとき−1が得られます。

■フローチャート



■プログラム例

```
10 INPUT X,Y
20 Z=X+Y
30 S=SGN(Z)
40 PRINT"X+Y=";S
50 GOTO 10
60 END
```

結果例		
X	Y	S
3	2	1
3	−5	−1
2	−2	0

〔例　題〕⑥

1～10の乱数をつくり表示するプログラムをつくりなさい。

乱数はRND　X　で与えられます。〔Random(ランダム)〕

RND　Xにおいて、Xの値により次のような乱数を得ることができます。

●Xが2以上の場合

Xが正の整数のとき：　1からXの値以下の乱数を発生します。
(1≦RND　X≦X)

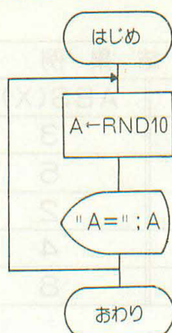
Xが小数部を含むとき：1からXの整数部に1を加えた値以下の乱数を発生します。
(1≦RND　X≦(INT　X)+1)

●Xが負数の場合　同じ乱数（あるいは乱数列）を発生させるために、初期値を一定にします。

● **Xが0以上、2未満の場合**　0より大きく、1より小さい値の乱数を発生します。

(0<RND　X<1)

■フローチャート



■プログラム例

```
5  WAIT 50
10 A=RND 10
20 PRINT"A=";A
30 GOTO 10
40 END
```

この部分の数値を

2.8 → 小数部	
−5 → 負　数	
0.5 → 1未満	

と変えて、乱数の出かたをいろいろ確認してください。

5　WAIT　50　は20行のPRINT命令の表示時間を指定しています。
ただし、一般のパソコンではWAIT指定ができないので次のようにします。

```
25 FOR J =1 TO 500:NEXT J
```

(FOR、NEXTについては104ページ参照)

〔例　題〕⑦

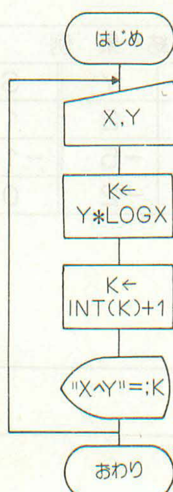
> 数XとYを入力してX^Yは何桁の数かを求めるプログラムをつくりなさい。
> ただし、XとYは正の整数値とします。

■解　説

常用対数を用います。

対数の性質から、Y・LOGXの整数値+1桁となります。

■フローチャート



■プログラム例

```
10 INPUT X,Y
20 K=Y*LOG X
30 K=INT(K)+1
40 PRINT X;"^";Y;"=";K;"ケタ"
50 GOTO 10
60 END
```

メモリ内容	
変　数	内　容
X	入力値
Y	入力値
K	桁　数

〔例　題〕⑧

> θに適当な数を入れsin, cos, tanの値を求めるプログラムをつくりなさい。
> (θ=30°, 45°, 60°と入力をしなさい。)

■解　説

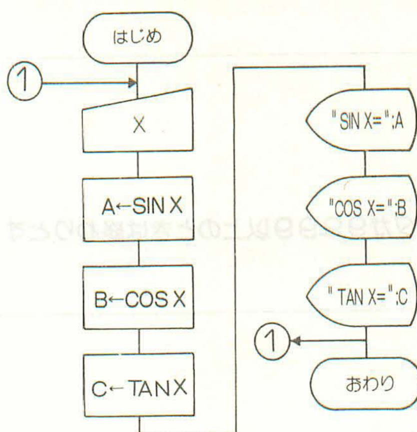
三角関数キーを用います。

角度単位を度(DEG)に指定します。

■プログラム例

```
5  DEGREE
10 INPUT X
20 USING
```

■フローチャート




```
30 A=SIN X:B=COS X:C=TAN X
40 PRINT "SIN" ;X; "="
   ;USING "##.####" ;A
50 USING
60 PRINT "COS" ;X; "="
   ;USING "##.####" ;B
70 USING
80 PRINT "TAN" ;X; "="
   ;USING "##.####" ;C
90 GOTO 10
100 END
```

メモリ内容	
変　数	内　容
X	入力値
A	SIN X
B	COS X
C	TAN X

〰〰 線で示す内容がある場合と、ない場合の表示のしかたの違いを確認してください。

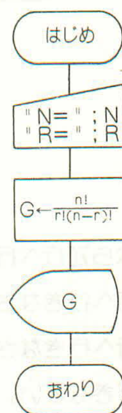
〔例　題〕⑨

組み合せ計算nCr$=\frac{n!}{r!(n-r)!}$の値を求めるプログラムをつくりなさい。

■解　　説

階乗(FACT)を用います。なお組み合せ(NCR)を用いれば簡単になります。

■フローチャート



■プログラム例

```
10 INPUT "N=";N, "R=";R
20 G=FACT N/(FACT R*FACT(N-R))
30 PRINT G
40 END
```

(注)20行はG=NCR(N,R)でも良い。

メモリ内容	
変　数	内　容
N	入力値
R	入力値
G	n!/r!(n−r)!

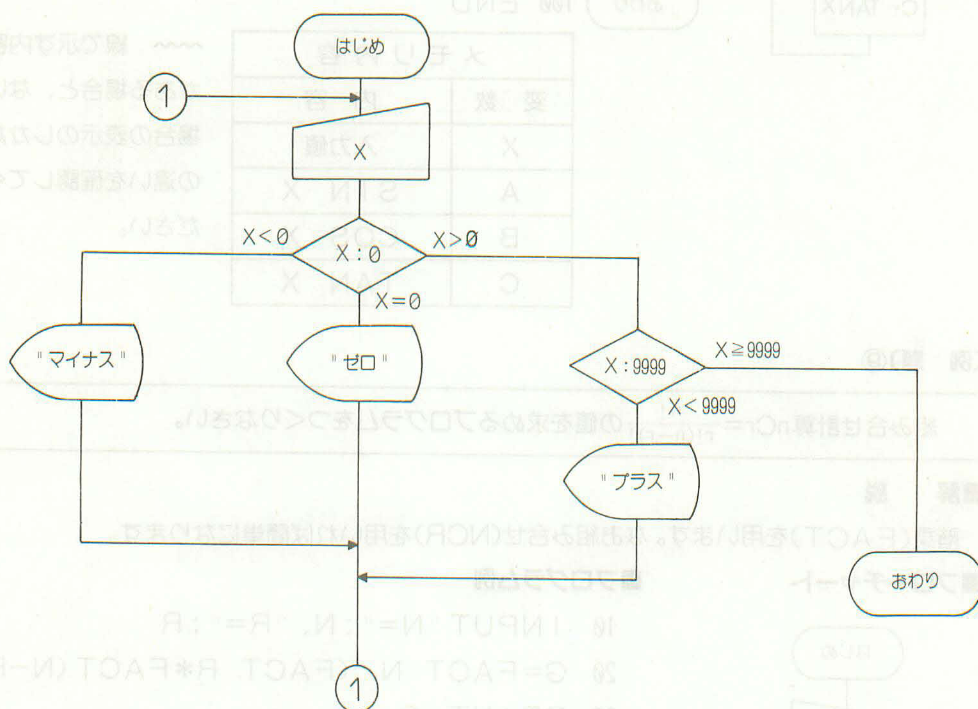
結　果　例		
n	r	nCr
5	3	10
10	4	210

STEP④ IF～THEN／IF～GOTO

〔例　題〕⑩

> 数Xを入力し、正の数なら"プラス　デス"
> 　　　　　　負の数なら"マイナス　デス"
> 　　　　　　0なら"ゼロ　デス"
> と表示するプログラムをつくりなさい。ただし、データが9999以上のときは終わりとするプログラムを考えなさい。

■フローチャート



■プログラム例

```
10  INPUT"X=";X
20  IF  X<0  THEN  70
30  IF  X=0  THEN  90
40  IF  X>=9999  THEN  110
50  PRINT"プラス␣デス"
60  GOTO  10
70  PRINT"マイナス␣デス"
80  GOTO  10
90  PRINT"ゼロ␣デス"
100 GOTO  10
110 END
```

（20行）もし、Xの値が負なら70行へ行きなさい。そうでないなら30行へ行きなさい。

（30行）Xの値が0なら90行へ行きなさい。そうでないなら40行へ行きなさい。

（40行）もし、Xの値が9999以上なら110行へ行きなさい。そうでないなら50行へ行きなさい。

■解　説

① **IF~THEN**はＩＦ以下の条件が成立するとき、ＴＨＥＮ以下の命令に従います。

　　㋑ＴＨＥＮ　文番号──→ その文番号へ行きます。

　　㋺ＴＨＥＮ　ステートメント──→ そのステートメントを実行した後、次の文番号へ行きます。

　　㋩条件が成立しないときは、次の文番号へ行きます。

② **GOTO　文番号**は、無条件で指定された文番号へ行きます。

③ **IF~THEN　文番号**は**IF~GOTO　文番号**と同じ命令です。

条　件　式	判　断　内　容	
○○＝××	等しいかどうか判断	（○○は××に等しいか？）
○○＞××	大きいかどうか判断	（○○は××より大きいか？）
○○＞＝××	以上かどうか判断	（○○は××以上か？）
○○＜××	小さいかどうか判断	（○○は××より小さいか？）
○○＜＝××	以下かどうか判断	（○○は××以下か？）
○○＜＞××	等しくないかどうか判断	（○○と××は等しくないか？）

〔例　題〕⑪

二次方程式 $ax^2+bx+c=0\ (a\neq 0)$の解の種類を判別して解を求めるプログラムをつくりなさい。ただし、２実根のときは＂２ジッコン＂、重根のときは＂ジュウコン＂、虚根のときは＂キョコン＂と表示させてそれぞれの解を求めなさい。

■解　説

一般式　$ax^2+bx+c=0$　（$a\neq 0$）の解の公式は

$$x=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

$$D=b^2-4ac$$

とすると

D＞0……………２実根 $\begin{cases} x_1=\dfrac{-b+\sqrt{D}}{2a} \\ x_2=\dfrac{-b-\sqrt{D}}{2a} \end{cases}$

D＝0……………重　根　$x=\dfrac{-b}{2a}$

D＜0……………虚　根

$D'=|D|$ とすれば、$\sqrt{-D'}$ ということです。$\sqrt{\ }$ の中が負になることは、数学的に（コンピュータが演算するうえでも）許されていません。

そこで $\sqrt{-D}$のことを$\sqrt{D}\cdot i$ と数学では表わしています。

$i=\sqrt{-1}$　ということです。

したがって、Dの絶対値を考えなければなりません。

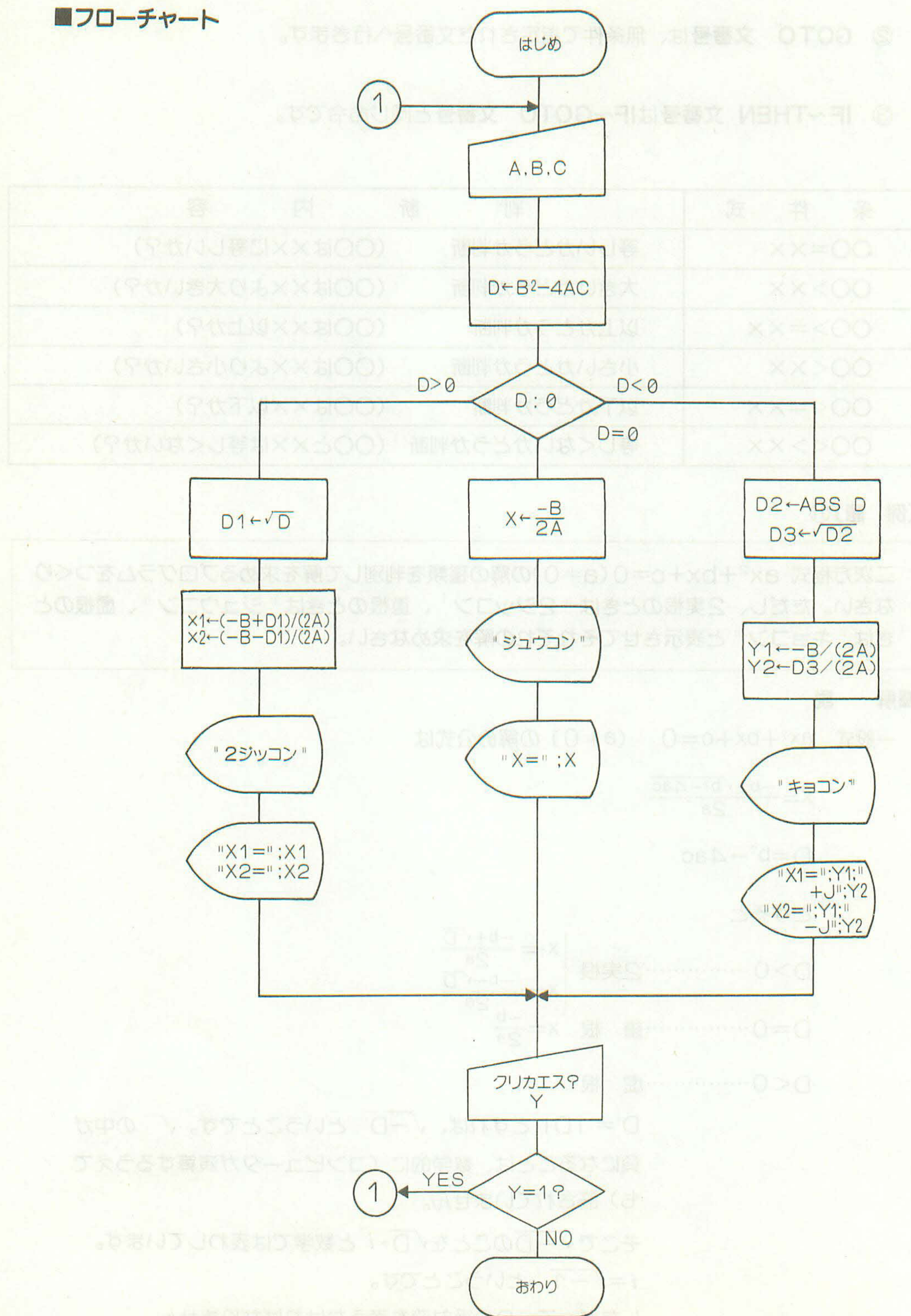
虚根は $x_1 = \frac{-b+\sqrt{D}i}{2a}$

$x_2 = \frac{-b-\sqrt{D}i}{2a}$

で表されます。

i のことは、虚数単位といいますが、本機の場合 i がありませんので、i の代わりにJを用いることにします。

■フローチャート



■プログラム例

```
10  INPUT"A=";A,"B=";B,"C=";C
20  D=B∧2-4*A*C
30  IF D<0 THEN 160
40  IF D=0 THEN 120
50  D1=SQR D
60  X1=(-B+D1)/(2*A)
70  X2=(-B-D1)/(2*A)
80  PRINT"2ジッコン"
90  PRINT"X1=";X1
100 PRINT"X2=";X2
110 GOTO 210
120 X=-B/(2*A)
130 PRINT"ジュウコン"
140 PRINT"X=";X
150 GOTO 210
160 D2=ABS D:D3=SQR D2
170 Y1=-B/(2*A):Y2=D3/(2*A)
180 PRINT"キョコン"
190 PRINT"X1=";Y1;"+J";Y2
200 PRINT"X2=";Y1;"-J";Y2
210 INPUT"クリカエス?␣(YES=1)";Y
220 IF Y=1 THEN 10
230 END
```

メモリ内容	
変数	内容
A	2次係数
B	1次係数
C	定数
D	B^2-4AC 判別式
D1	$\sqrt{D}$
D2	ABS D
D3	$\sqrt{D2}$
X	−B/2A
X1	(−B+D1)/2A
X2	(−B−D1)/2A
Y1	−B/2A
Y2	D3/2A
Y	繰り返し判断用

下表の数値を入れて結果を求めてください。

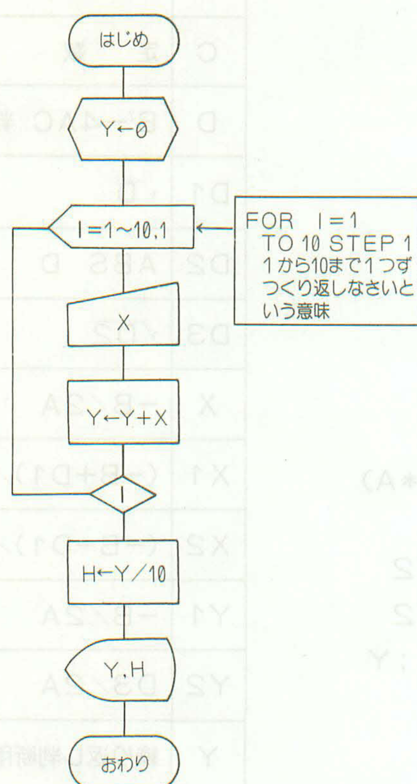
A	B	C	結果
1	5	6	2ジッコン X1=−2 X2=−3
1	2	1	ジュウコン X=−1
5	−10	25	キョコン X1=1+J2 X2=1−J2
1	1	$-3+\sqrt{3}$	2ジッコン X1=7.320508075×10^{-1} X2=−1.732050808
1	−3	6	キョコン X1=1.5+J1.936491673 X2=1.5−J1.936491673
2	$2\sqrt{10}$	5	ジュウコン X=−1.58113883

STEP⑤ FOR～TO～STEP,NEXT

〔例　題〕⑫

数値を10個入力して合計と平均を求めるプログラムをつくりなさい。

■フローチャート



■プログラム例

```
10 Y=0
20 FOR I=1 TO 10 STEP 1
30 INPUT"X=";X
40 Y=Y+X
50 NEXT I
60 H=Y/10
70 PRINT"ゴウケイ␣";Y
80 PRINT"ヘイキン␣";H
90 END
```

（20のIと50のI：対応しています）

メモリ内容	
変数	内　容
Y	初期値設定 Y＝0 累　計 Y＋X
X	入力値
H	平　均 Y／10

■解　　説

①文番号10のY＝0は変数Yの中にはなにも数値が入っていないという状態をつくっています。
これを最初につくっておかないと、文番号40のところで、誤った値を加算することになります。

②文番号20のＳＴＥＰ　1はこの値が1の場合に限って省略してもよいのです。
ＦＯＲ　ＩのＩとＮＥＸＴ　ＩのＩは同じ文字にしなければなりません。
ただし、ＮＥＸＴ　ＩのＩは省略できます。
たとえば、ＦＯＲ　Ｊ＝１　ＴＯ　10とすれば、ＮＥＸＴ　Ｊとするか、ＮＥＸＴということになります。
しかし、慣れるまではＮＥＸＴの後の文字を省略しないほうがよいでしょう。
プログラムを見直すときは、ＮＥＸＴの後の文字があるほうが便利です。

③文番号40のＹ＝Ｙ＋Ｘは累計を求めています。

④文番号60のＨ＝Ｙ／１０は平均を求めています。

■累計の数値例

Iの値	Y+X	Y=Y+Xの値
1	0+1	1
2	1+2	3
3	3+3	6
4	6+4	10
5	10+5	15
6	15+6	21
7	21+7	28
8	28+8	36
9	36+9	45
10	45+10	55

●左の表のようにYの値はIの値の変化とともにそれぞれ累計されていきます。

FOR~NEXT文を使用しないで合計を求めるプログラム

〔例題〕⑫で、FOR~NEXT文を使用しない足し込みの方法は、

■プログラム例

```
10 A=0
20 Y=0
30 INPUT"X=";X
40 A=A+1
50 Y=Y+X
60 IF A<10 THEN 30
70 H=Y/10
80 PRINT"ゴウケイ=";Y
90 PRINT"ヘイキン=";H
100 END
```

①このプログラムは30行と60行の間を条件が成立するまで、くり返しを行っていることになります。(くり返しループを形成します)

②40行のA=A+1は左辺のAが10になるまで60行のIF文で判断され、10回のループの足し込みを行います。

③そのAの足し込みの中でのもうひとつの足し込みが50行のY=Y+Xで、Xの値が左辺のYの変数に累計されます。Aが10になった時点で累計完了ということになります。

「FOR~NEXT」文の二重ループ

〔例　題〕⑬

1×5，1×10，1×15 2×5，2×10，2×15 ⋮ 5×5，5×10，5×15	をそれぞれ表示し、答えも表示するプログラムをつくりなさい。

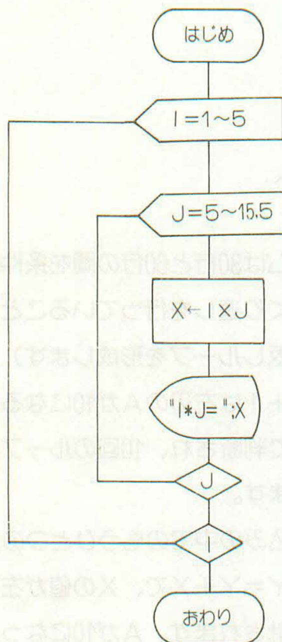
■解　説①

1～5　の1つ刻みのループ（STEP　1）と

5～15　の5つ刻みのループ（STEP　5）

がありますので、FOR～NEXTの二重ループとなります。

■フローチャート



■プログラム例

```
10 FOR I=1 TO 5
20 FOR J=5 TO 15 STEP 5
30 X=I*J
40 PRINT I;"*";J;"=";X
45 FOR K=1 TO 500:NEXT K
50 NEXT J
60 NEXT I
70 END
```

■解　説②

文番号20と50が対応し、文番号10と60が対応しています。IとJの値について、ループ内での数値の動きは

I=1⟶J=5，J=10，J=15
　となり、ここまで来てI=2に移ります。

I=2⟶J=5，J=10，J=15
　となり、ここまで来てI=3に移ります。

I=3⟶J=5，J=10，J=15
　となり、ここまで来てI=4に移ります。

I=4⟶J=5，J=10，J=15
　となり、ここまで来てI=5に移ります。

I=5⟶J=5，J=10，J=15
　となり、ここでプログラムは終了となります。

■FOR〜NEXTについて

例−1 下のプログラム例のようにループ①がループ②の中に完全に入っている使い方が正しい使い方です。

このポケットコンピュータでは五重までのループが可能です。

```
10  FOR  I=1  TO  5 ─────────┐
 :                             │
40  FOR  J=1  TO  1Ø ──┐       │
 :                     │ループ①│ループ②
80  NEXT  J ───────────┘       │
 :                             │
100 NEXT  I ───────────────────┘
```

例−2

```
10  FOR  I=1  TO  5 ─────────┐
 :                             │
40  FOR  J=1  TO  1Ø ──┐       │ループ②
 :                     │ループ① │
80  NEXT  I ───────────┼───────┘
 :                     │
100 NEXT  J ───────────┘
```

このような、プログラムはERRORとなります。ループ①とループ②が交差することは文法上許されません。

ERROR 52が生じます。

例−3

```
40  GOTO  8Ø
 :
60  FOR  B=−1Ø  TO  12  STEP  2 ──┐
 :                                 │
80  D=B∧2−4*A                      │ループ
 :                                 │
100 NEXT  B ───────────────────────┘
```

外からFOR〜NEXTループ内にとび込むことはできません。

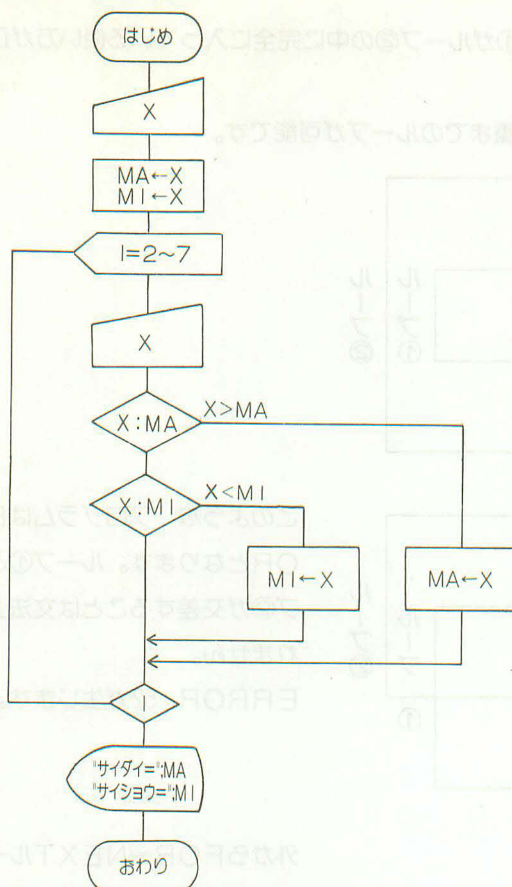
〔例 題〕⑭

7つのデータがあります。15、21、36、81、9、16、10をそれぞれ入力し、最大値と最小値を求めるプログラムをつくりなさい。

■プログラム例

```
10  INPUT"X=";X
20  MA=X:MI=X
30  FOR  I=2  TO  7
40  INPUT"X=";X
50  IF  X>MA  THEN  MA=X
60  IF  X<MI  THEN  MI=X
70  NEXT  I
80  PRINT"サイダイ=";MA;"サイショウ=";MI
90  END
```

■フローチャート



メモリ内容	
変数	内容
MA	最大値
MI	最小値
X	データ入力値

■解　説

①文番号10で最初のデータを入力します。

②文番号20の　MA＝X：MI＝Xで、最初の入力データを仮に最大値、最小値にします。

③文番号30〜70で、データをひとつずつ入力していきます。

④文番号50〜60で、最大値、最小値の比較を行っています。

下の表で最大値、最小値を比較している内容を細かく分析してみます。

Xの値	最大値の比較 X>MA	最大値	最小値の比較 X<MI	最小値
15のとき	—	15となる	—	15となる
21のとき	21>15	21に変る	21<15	15のまま
36のとき	36>21	36に変る	36<15	15のまま
81のとき	81>36	81に変る	81<15	15のまま
9のとき	81>9	81のまま	9<15	9に変る
16のとき	16>81	81のまま	16<9	9のまま
10のとき	10>81	81のまま	10<9	9のまま

↑ 最終結果・最大値　　↑ 最終結果・最小値

STEP⑥
REM, READ, DATA, RESTORE

〔例　題〕⑮

50、70、80、65、50を読んで、平均を求めるプログラムをつくりなさい。ただし、ヘイキンという注釈文をプログラムリストの最初につけなさい。

■解　　説

①注釈文をつけるときは、REM文を用います。(REM　または，)

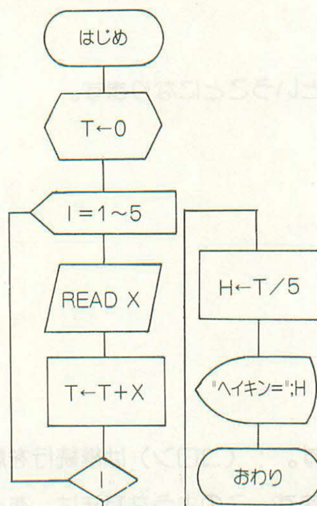
REMは、注釈の意味でプログラムの実行には、関係ありません。しかし、リストをみるときや、リストを印字するときは、どのようなプログラムなのか、ということがわかります。また、長いプログラムなどの場合においては、演算処理の項目ごとに、REM文で注釈をつけておきますと、あとでプログラムリストを検討するとき、大変役に立つ命令文となります。

②REM文は、任意の文番号をつけることができます。

③READ文は、DATA文と対になって変数にデータを割りあてます。

④対応するデータがなくなると、ERROR 53が生じます。

■フローチャート



■プログラム例

```
10  REM ヘイキン
20  T=0
30  FOR I=1 TO 5
40  READ X
50  T=T+X:REM ゴウケイ
60  NEXT I
70  H=T/5
80  PRINT"ヘイキン=";H
90  END
100 DATA 55,70,80,65,50
```

■結　　果

ヘイキン=64.

■プログラムの解説

10　ヘイキン　という注釈文になります。　10'ヘイキン　としてもよいです。

20　Tの変数を0（ゼロ）にします。ここに、CLEAR命令を用いてもよいです。

30　30行から60行のFOR～NEXTループです。Ｉの値が1から始まって、5を超えるまでくり返します。

40　READ　文です。

READ　Xは、100行のDATA文の最初のデータ55を読み込みます。

50　足し込みを行っています。このときのTは、T=T+X（↙20行の0（ゼロ）、55↗、└DATA 55）

60　NEXT　Iで30行にもどります。このとき、すでに、I=2となっています。

40行のREAD　Xは、100行のDATA文の70を読み込みます。

50行は　T=T+X　←DATA 70
　125┘　└I=1のときのTの合計 55

となります。60行のNEXT　Iでまた30行にもどります。

このときのIの値は、I=3となっています。

40行のREAD　Xは、100行のDATA文の80を読み込みます。

50行は　T=T+X　←DATA 80
　205┘　└I=2のときのTの合計 125

となります。60行のNEXT　Iでふたたび30行にもどります。

Iの値は、I=4となっています。

40行のREAD　Xは、100行のDATA文の65を読み込みます。

50行は　T=T+X　←DATA 65
　270┘　└I=3のときのTの合計 205

となります。60行のNEXT　Iで、もう一度30行にもどります。

Iの値は、I=5となり、40行のREAD　Xは、100行のDATA文の50を読み込んで

50行は　T=T+X　←DATA 50
　320┘　└I=4のときのTの合計 270

となります。したがって、5つのデータの合計は、320ということになります。

70　H=T/5（┌320が入ります）

よって、変数Hには、64という数字が入ります。

80　Hの値を表示します。

90　プログラムの実行が終了します。

100　DATA文です。

★★★★★

文番号50の：REM　ゴウケイ　は、やはり注釈文となります。：（コロン）は継続行を意味しますから、こんなかたちで、REM文を使うこともできます。このような方法は、あとでプログラムを解析するとき、実に効果的なものとなります。

〔例　題〕⑯

> 下のデータを用いて、A=60、B=50、C=10、D=40、E=20、F=30、G=60、H=50となるように、データを読み込むプログラムをつくりなさい。
>
> 60、50、10、40、20、30

■解　説

RESTORE　文番号は、データを再び文番号の先頭から読み込む命令文です。

同じデータを何回も使用するときは、効果的な手段となります。

■フローチャート

■プログラム例

```
10  REM  RESTORE
20  READ  A, B
30  READ  C, D, E, F
40  RESTORE  200
50  READ  G, H
60  PRINT"A=";A;"␣B=";B
70  PRINT"C=";C;"␣D=";D
80  PRINT"E=";E;"␣F=";F
90  PRINT"G=";G;"␣H=";H
100 FOR  J=1  TO  1000:NEXT  J
110 END
200 DATA  60, 50
210 DATA  10, 40, 20, 30
```


■結　　果

```
A=60  B=50
C=10  D=40
E=20  F=30
G=60  H=50
```

■プログラムの解説

10　注釈文です。

20　READ　AのAは、200行のDATA文にある60を読み込み、続いてBは、50を読み込みます。

30　READ　C，……，Fは、210行の、10、40、20、30をそれぞれ順次読み込んで行きます。

40　200行のDATA文を再び呼び出します。

50　そして、READ　GのGには、200行のDATA文の60、Hには、50を読み込ませます。

60～110　それぞれ、読み込んだデータを表示します。110行でプログラムの実行は終ります。

なお、100行は画面をしばらく止めておくために入れています。

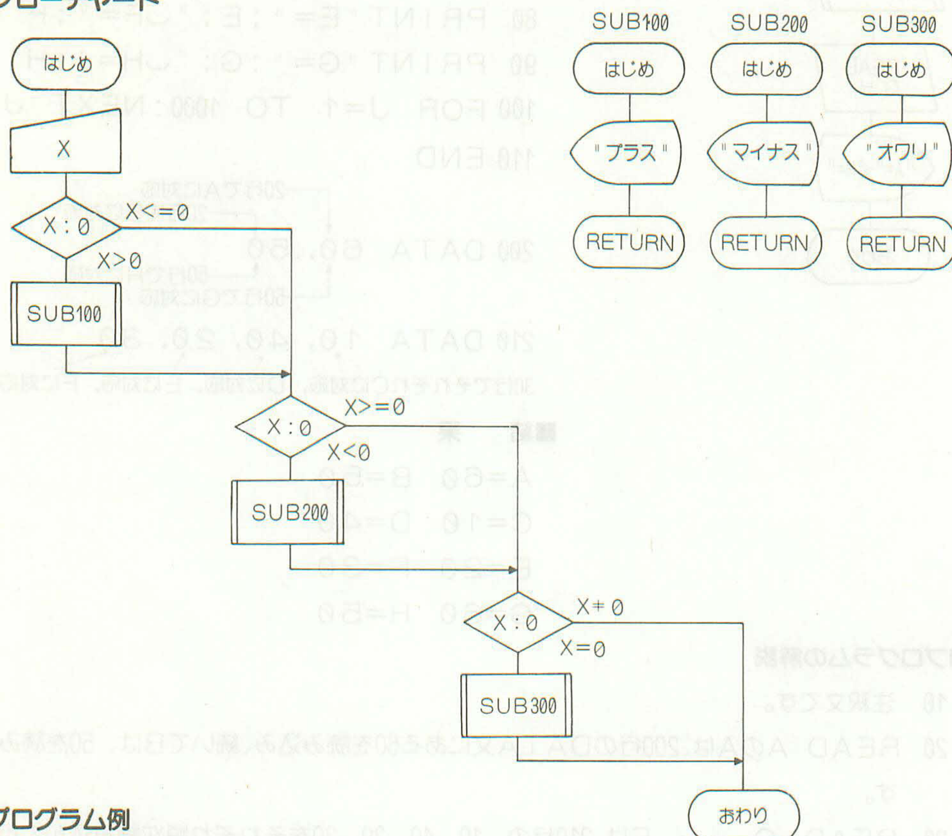
★★★　200行と210行は、DATA文です。

STEP ⑦ GOSUB~RETURN

〔例　題〕⑰

数Xを入力してプラスのときは"プラス"、マイナスのときは"マイナス"、0のときは"オワリ"と表示するプログラムをつくりなさい。

■フローチャート



■プログラム例

```
10  INPUT  X
20  IF  X>0  THEN  GOSUB100
30  IF  X<0  THEN  GOSUB200
40  IF  X=0  THEN  GOSUB300
50  END
100 PRINT "プラス"
110 RETURN
200 PRINT "マイナス"
210 RETURN
300 PRINT "オワリ"
310 RETURN
```

■解　説

①独立した1つのプログラムで、いつでも呼び出して使えます。

②呼び出し方は **GOSUB 文番号**となります。

③サブプログラムの終わりはRETURNです。

④このプログラムの実行順序はX＝－3とすれば、10→20→30→200→210→40→50です。

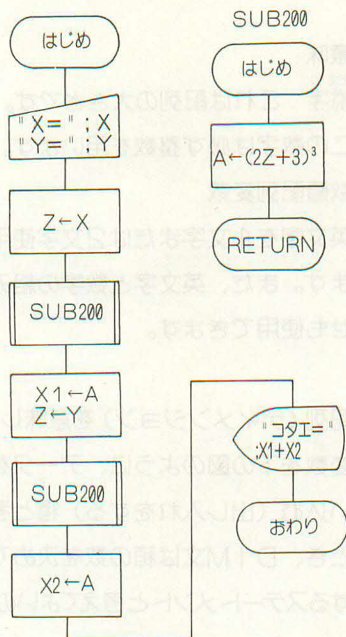
〔例　題〕⑱

> 数XとYを入力し、$(2X+3)^3+(2Y+3)^3$を求めるプログラムをつくりなさい。

■解　　説

$(2X+3)^3$と$(2Y+3)^3$はXとYをZとおきかえれば同じ形です。

■フローチャート



■解　　説

①$(2X+3)^3$と$(2Y+3)^3$はXとYをZとおくと同じ形となり、サブプログラムで計算できます。

②$(2Z+3)^3$を求めるサブプログラムで$(2X+3)^3$を求めるためにZ=Xとして、Xの値をZに代入しておきます。（ここが重要です。）

③サブプログラムの計算結果はX1=AとしてX1に代入しておきます。

同じようにX2=AとしてX2に代入しておきます。

■プログラム例

```
10  INPUT"X=";X
20  INPUT"Y=";Y
30  Z=X
40  GOSUB 200
50  X1=A
60  Z=Y
70  GOSUB 200
80  X2=A
90  PRINT"コタエ=";X1+X2
100 END
200 A=(2*Z+3)∧3
210 RETURN
```

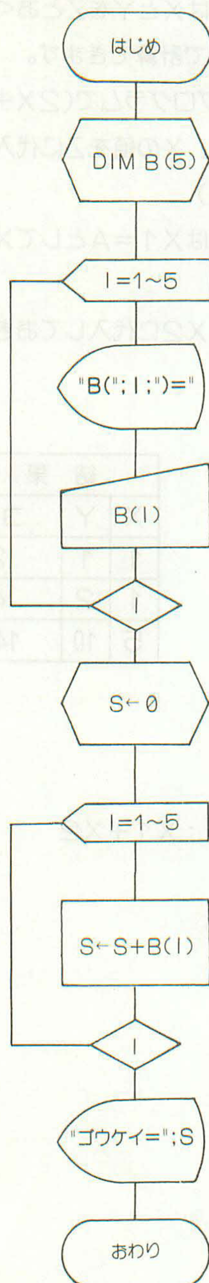
結果例		
X	Y	コタエ
1	1	250
1	2	468
5	10	14364

STEP ⑧ 配列DIM(ディメンジョン)

〔例　題〕⑲

> 5個の数　25、18、23、17、9を読み込んで、読み込んだ順に配列変数Bに代入し、次に5個の数の合計を求めて表示するプログラムをＤＩＭ文を用いてつくりなさい。

■フローチャート

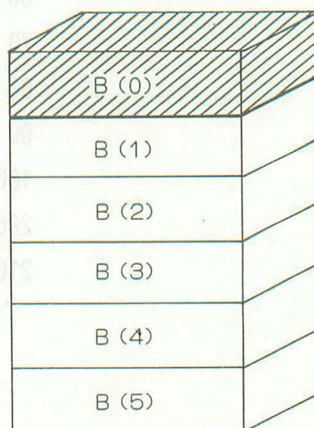


1次元配列

ＤＩＭ　Ｂ（５）の意味

- 添字（そえじ）　これは配列の大きさです。
- この数字は必ず整数を用います。
- 数値配列変数
  英文字を１文字または２文字使用できます。また、英文字と数字の組み合わせも使用できます。
- 配列（ディメンジョン）を意味します。
  変数を下の図のように、データをしまい込む（出し入れをする）箱と考えたとき、ＤＩＭ文は箱の数を決めて用意するステートメントと考えてよいのです。

ＤＩＭ　Ｂ（５）は



- B(0)からB(5)までB(0)を含めて、6個の箱が用意されます。
- この例題のプログラム例では斜線の部分は未使用ということになります。

■プログラム例

```
10 DIM B(5)
20 FOR I=1 TO 5
30 PRINT"B(";I;")=";
40 INPUT B(I)
50 NEXT I
60 S=0
70 FOR I=1 TO 5
80 S=S+B(I)
90 NEXT I
100 PRINT"ゴウケイ=";S
110 END
```

結　果　例
B(1)=25
B(2)=18
B(3)=23
B(4)=17
B(5)=9
ゴウケイ=92

このようにB(I)のIには、順次1、2、3、4、5という数字が代入されていきます。

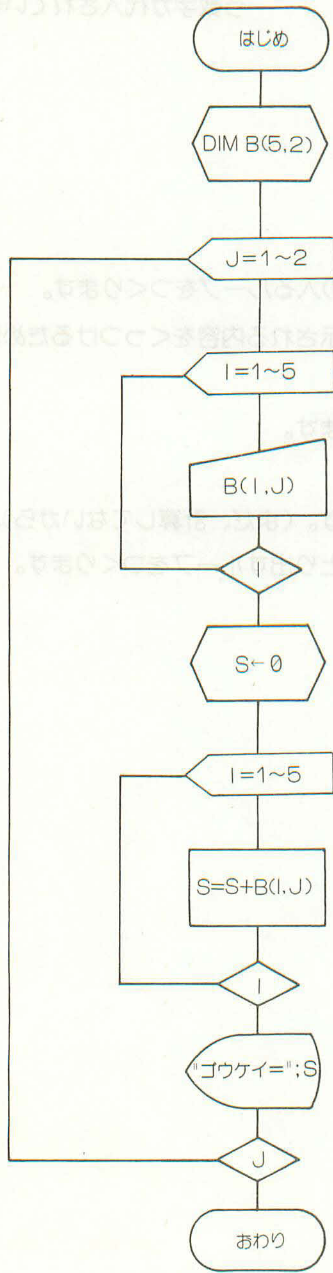
■プログラムの解説

10　配列を指定します。
20　1から5まで（入力する値が5個ですから）の数値の入るループをつくります。
30　入力する順番を表示させます。最後の；は40行で表示される内容をくっつけるために入れています。
40　数値を入力します。30行で表示した内容の後に入ります。
50　5個入力するまで20行へ行きます。
60　合計を求める前に今の合計値は0ですと指定をします。（まだ、計算してないから）。
70　B(I)の箱の中に入れた5個の数値を1から5までとり出すループをつくります。
80　ひとつずつとり出したら、それぞれ累計します。
90　5個B(I)からとり出すまで70行へ行きます。
100　5個とり出し終えて、累計した結果を表示します。
110　プログラムは終わりです。

〔例　題〕⑳

> 右の図のように5個の数が2列あります。
> 列ごとの合計を求めて表示するプログラムを
> つくりなさい。
>
> | | | |
> |---|---|---|
> | | 25 | 38 |
> | | 18 | 46 |
> | | 23 | 92 |
> | | 17 | 73 |
> | | 9 | 65 |
> | 合　計 | | |

■フローチャート



2次元配列

ＤＩＭ　Ｂ(5，2)

　　　(行，列)

　　　5行，2列の配列を意味します。

ＤＩＭ　Ｂ(5，2)の2次元配列は下の図のような箱が用意されると考えられます。

B(0,0)	B(0,1)	B(0,2)
B(1,0)	B(1,1) 25	B(1,2) 38
B(2,0)	B(2,1) 18	B(2,2) 46
B(3,0)	B(3,1) 23	B(3,2) 92
B(4,0)	B(4,1) 17	B(4,2) 73
B(5,0)	B(5,1) 9	B(5,2) 65

例題⑳では斜線の部分は未使用ということになります。

■プログラム例

```
10 DIM B(5, 2)
20 FOR J=1 TO 2
30 FOR I=1 TO 5
40 INPUT B(I, J)
50 NEXT I
60 S=0
70 FOR I=1 TO 5
80 S=S+B(I, J)
90 NEXT I
100 PRINT "ゴウケイ=";S
110 NEXT J
120 END
```

メモリ内容	
変数	内　容
B	B(5,2) 二次元配列 B(I,J)
I	I=1〜5
J	J=1〜2
S	合　計 S=0 S=S+B(I,J)

■プログラム例の解説

10 2次元配列、5行・2列を指定します。

20 Jを1から2と順次変化させます。しかし30行目のIが5になり終わるまではJ=1を維持しつづけます。

30 Iを1から5と順次変化させます。

40 B(I, 1) このIは1から5と変化していきます。

50 30行のIと対となって順次Iの値を変化させます。

60 合計を求めるために、まだ計算前ですのであらかじめS=0としておきます。

70 合計を求めるために5個入力した数値を順次呼び出しはじめます。

80 合計を求めるために順次累計をしはじめ、B(5, 1)で終わります。

90 70行のIと対となってB(5, 1)になるまで順次Iの値を変化させます。

100 ゴウケイ=と表示し、合計した値を表示します。

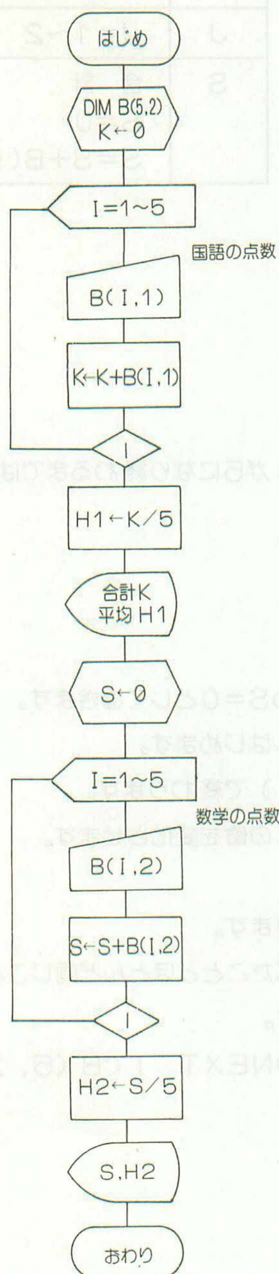
110 20行にもどり、ここではじめてJ=2という値になります。
プログラムの流れはこれより30行に行き、上記に述べたこととほとんど同じことをします。
ただ、40行のB(I, J)はB(I, 2)と変ります。
そして、さらに80行ではB(5, 2)となり、90行のNEXT IでB(5, 2)までとなり、100行でもうひとつの合計を表示します。

〔例　題〕㉑

下表のような５人の生徒の成績があります。科目ごとの合計と平均点を計算するプログラムをつくりなさい。

番号	1	2	3	4	5
国語	58	45	69	87	75
数学	45	25	78	82	65

■フローチャート



■プログラム例

```
10 DIM B(5, 2)
20 K=0
30 FOR I=1 TO 5
40 INPUT"コクゴーテンスウ=";B(I, 1)
50 K=K+B(I, 1)
60 NEXT I
70 H1=K/5
80 PRINT"ゴウケイ=";K;"ヘイキン=";H1
90 S=0
100 FOR I=1 TO 5
110 INPUT"スウガクーテンスウ=";B(I, 2)
120 S=S+B(I, 2)
130 NEXT I
140 H2=S/5
150 PRINT"ゴウケイ=";S;"ヘイキン=";H2
160 END
```

メモリ内容	
変数	内　容
B( )	国語の点数　数学の点数
H1	H1=K/5 国語の平均
H2	H2=S/5 数学の平均
K	K=0　K=K+B(I,1) 国語の合計点
S	S=0　S=S+B(I,2) 数学の合計点
I	I=1～5

〔例　題〕㉒

下のような数値の表があります。

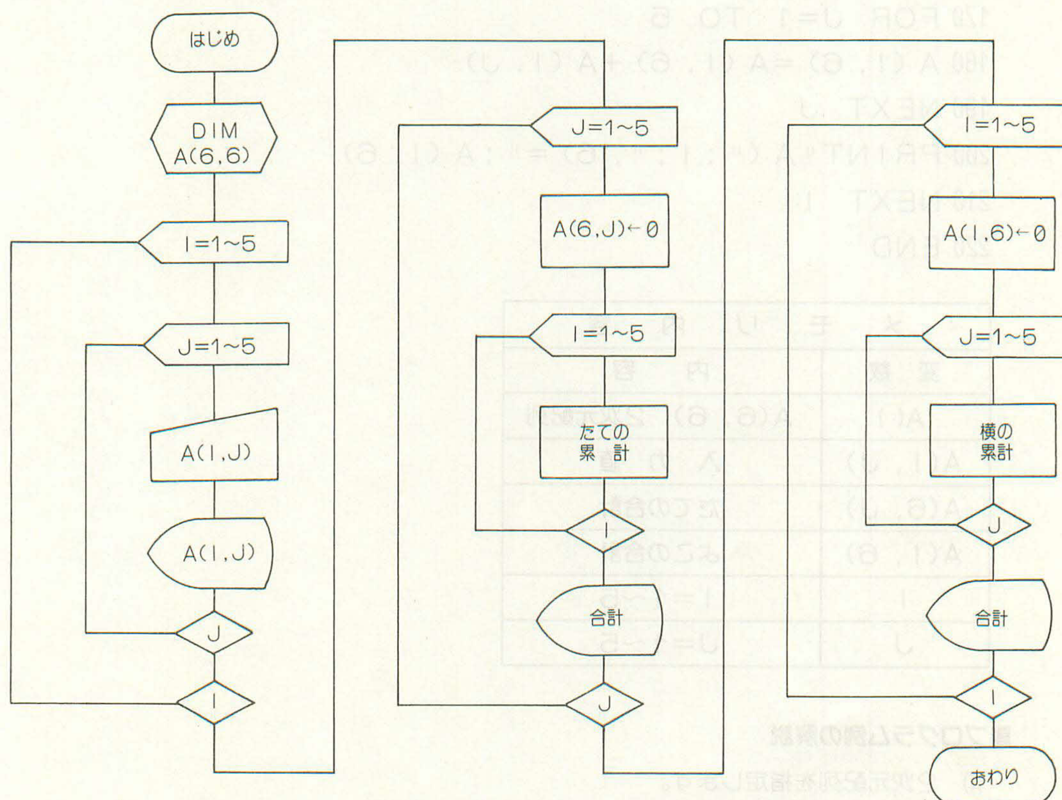
数値を、1、2、3、4、5

11、12、13、14、15

と順次入力していき、たて、横の合計を求めるプログラムをつくりなさい。

						横の合計
	1	2	3	4	5	
	11	12	13	14	15	
	21	22	23	24	25	
	31	32	33	34	35	
	41	42	43	44	45	
たての合計						

■フローチャート



■プログラム例

```
10 DIM  A(6, 6)
20 FOR  I=1  TO  5
30 FOR  J=1  TO  5
40 INPUT  A(I, J)
50 PRINT"A(";I;",";J;")=";A(I, J)
60 NEXT  J
70 NEXT  I
80 FOR  J=1  TO  5
90 A(6, J)=0
100 FOR  I=1  TO  5
110 A(6, J)=A(6, J)+A(I, J)
120 NEXT  I
130 PRINT"A(6,";J;")=";A(6, J)
140 NEXT  J
150 FOR  I=1  TO  5
160 A(I, 6)=0
170 FOR  J=1  TO  5
180 A(I, 6)=A(I, 6)+A(I, J)
190 NEXT  J
200 PRINT"A(";I;", 6)=";A(I, 6)
210 NEXT  I
220 END
```

メ モ リ 内 容	
変 数	内 容
A( )	A(6, 6) 2次元配列
A(I, J)	入 力 値
A(6, J)	たての合計
A(I, 6)	よこの合計
I	I=1〜5
J	J=1〜5

■プログラム例の解説

10　2次元配列を指定します。

20　20行から70行までは、データを入力し表示するループです。

90　A(6, J)=0はたての合計ですが、まだ計算前ですから0を代入します。

160　A(I, 6)=0はよこの合計ですが、まだ計算前ですから0を代入します。

40行、50行目について、実行例で説明をします。

```
RUN ↵          ?
1   ↵          A(1. , 1.)=1.
               ?
```

```
2    ↵    A(1., 2.)=2.
          ?
3    ↵    A(1., 3.)=3.
          ?
4    ↵    A(1., 4.)=4.
          ?
5    ↵    A(1., 5.)=5.
          ?
11   ↵    A(2., 1.)=11.
          ?
12   ↵    A(2., 2.)=12.
          ?
```

と、このように、何行何列目にどのような数値が入ったかを確認できるようにしたものです。

110行目の、たての累計計算は下に示すような形でコンピュータは計算しています。

A(6,J)=A(6,J)+A(I,J)		J=1のとき	J=2のとき	J=3のとき	J=4のとき	J=5のとき
たての累計計算の途中結果	はじめ	A(6,1)=0	A(6,2)=0	A(6,3)=0	A(6,4)=0	A(6,5)=0
	Iは1	A(6,1)=1	A(6,2)=2	A(6,3)=3	A(6,4)=4	A(6,5)=5
	Iは2	A(6,1)=12	A(6,2)=14	A(6,3)=16	A(6,4)=18	A(6,5)=20
	Iは3	A(6,1)=33	A(6,2)=36	A(6,3)=39	A(6,4)=42	A(6,5)=45
	Iは4	A(6,1)=64 ①	A(6,2)=68 ②	A(6,3)=72 ③	A(6,4)=76 ④	A(6,5)=80 ⑤
最終結果(合計)	Iは5	A(6,1)=105	A(6,2)=110	A(6,3)=115	A(6,4)=120	A(6,5)=125

A(6, J)=A(6, J)+A(I, J)の算術式のIとJでは、くりかえしループ(FOR~NEXT文)によって上の表のような順序 ①→②→③→④→⑤ で計算がおこなわれています。

180行目の横の累計計算も同様に、下に示すような形でコンピュータは計算しています。

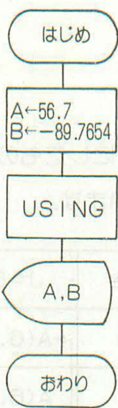
A(I,6)=A(I,6)+A(I,J)		I=1のとき	I=2のとき	I=3のとき	I=4のとき	I=5のとき
横の累計計算の途中結果 ①	はじめ	A(1,6)=0	A(2,6)=0	A(3,6)=0	A(4,6)=0	A(5,6)=0
	Jは1	A(1,6)=1	A(2,6)=11	A(3,6)=21	A(4,6)=31	A(5,6)=41
	Jは2	A(1,6)=3	A(2,6)=23	A(3,6)=43	A(4,6)=63	A(5,6)=83
	Jは3	A(1,6)=6	A(2,6)=36	A(3,6)=66	A(4,6)=96	A(5,6)=126
	Jは4	A(1,6)=10 ①	A(2,6)=50 ②	A(3,6)=90 ③	A(4,6)=130 ④	A(5,6)=170 ⑤
最終結果(合計)	Jは5	A(1,6)=15	A(2,6)=65	A(3,6)=115	A(4,6)=165	A(5,6)=215

STEP ⑨ USING（ユージング）PRINT USING

〔例　題〕㉓

A=56.7、B=−89.7654 の2つの数値において、それぞれ小数点以下第3位で桁をそろえて表示するプログラムをつくります。

■フローチャート



■プログラム例

```
10 A=56.7, B=−89.7654
20 USING "####.###"
30 PRINT A;B
40 END
```

20行で桁数の指定をします。

■実行手順

RUN モード

RUN ENTER

表示例

56.700 −89.765

〔例　題〕㉔

B=−10、C=50.8803 の2つの数値において文字と数値をそれぞれ表示するようにプログラムを考えなさい。小数点のあるCについては小数点以下2桁まで表示するものとします。

■フローチャート（省略）

■プログラム例

```
10 B=−10, C=50.8803
20 PRINT USING"&&&###";"B=";B;"␣C=";USING"###.##";C
30 END
```

実行表示例

B= −10 C= 50.88

（注意）

桁数指定を解除したいときは、解除したい行にUSINGのみ指定すればその行以降は桁数指定が解除されます。

STEP 10 MID$(ミッド・ドル)……中間の文字をとり出す
LEN(レングス)……文字列の文字を数える
VAL(バリュー)……文字から数字へ

〔例　題〕㉕

2進数を10進数に変換するプログラムを作りなさい。

■解　説

今までは数値のみを扱ってきましたが、これから文字列に関する練習をやってみましょう。コンピュータは数字か"文字"かはっきり指示してやらなければ動きませんが、逆に言うと人間にはできないこともやってくれます。

たとえば、2進数の10010110は10進数ではいくつになるでしょうか。

```
1   0   0   1   0   1   1   0
‖           ‖       ‖   ‖
2^7 +       2^4  +  2^2+ 2^1
128 +       16   +  4 + 2   =150
```

このような演算を上の桁から実行し、加算することにします。

■プログラム例

```
 10:"A":CLEAR:
    DIM B$(0)
 20:INPUT B$(0)
 30:L=LEN B$(0)
 40:N=1
 50:C$=MID$(B$(0),
    N,1)
 60:E=VAL C$
 70:D=E*2∧(L−1)
    +D
 80:N=N+1:L=L−1
 90:IF L=0 THEN
    110
100:GOTO 50
110:PRINT D
120:GOTO 10
```

10 最初にDIM B$(0)と配列宣言をして、代入する文字数を16に設定します。つまりいま入れられる2進数は最大16桁までになります。単にB$ですと最大7文字になります。

以下$(ドル)のついた変数が文字変数です。

30 LEN B$(0)は文字の数を数えます。10010110なら文字の数は8です。

50 C$=MID$(B$(0), N, 1)

B$(0)の文字列の　左からN番目の文字から　1文字とり出して　C$の中へ入れます。

60 VAL C$では文字(C$)を数字に変換して次の計算に備えます。1か0です。

80 Nはパラメーターです。1つずつ加えて行けば、B$(0)の文字列の左から1つずつ動かすことになります。L=0で最後の桁ですから計算を止めて、110ラインで答を表示します。

11111111=255. 1111111111111111=65535　となることを確認してください。

STEP⑪ CHR$(キャラクタドル)

数をアスキー文字に変える

STR$(ストリングドル)

数を文字に変える

〔例　題〕㉖

10進数を2進数に変換するプログラムをつくりなさい。

■解　説

10進数を2で割っていき、その余りを算出順に下位から順に並べて行くと2進数になります。たとえば10進数の43では、

```
       余り
2)43……1
2)21……1
2)10……0
2) 5……1
2) 2……0
2) 1……1
   0
(答)  1 0 1 0 1 1
```

■プログラム例

```
200：“B”：CLEAR ：
     DIM B$(0)
210：INPUT N
220：I=INT(N/2)
230：J=N-2*I
240：B$(0)=CHR$(
     J+48)+B$(0)
250：IF I>0 THEN
     N=I：GOTO 220
260：PRINT B$(0)
270：GOTO 200
```

200 例題㉕と同じくDIMB$(0)と配列宣言して、答の2進数が16桁まで表示できるようにします。

220 Nは2で割っていきますから1回ごとに変ります。

230 Jは余りです。1か、0です。

240 CHR$は、ASC(アスキー：次ページ参照)の逆で、10進数のアスキーコードを文字や記号（0～9の数字を含みます）に変換します。
CHR$(J+48)のJは0か1ですから、この答は“文字”の“0”か“1”が入ることになります。
本機では、CHR$、ASC（アスキー）が使えるので大いに活用してください。

250 Iが0になるまでNに代入して220ラインにもどり、各桁の余りをB$(0)に入れていき、文字列を完成させます。

215 に R$=STR$N を追加し、

260 を PRINT R$；“－>”；B$(0)としてみてください。

N（数）が文字に変って答に表示されます。

このプログラムでは最大で65535=1111111111111111　までです。

アスキー(ASCII)文字コード表

10進数	文　字
⋮	⋮
48	0
49	1
⋮	⋮
54	6
⋮	⋮
65	A
66	B
67	C
68	D
69	E
70	F
⋮	⋮

STEP⑫ ASC(アスキー)…文字を数で表す
論理演算子……………………ホント(真)かウソ(偽)か?

〔例　題〕㉗

> 10進数を16進数に、また16進数を10進数に変換するプログラムを作りなさい。

■解　　説

本機は&Hh(hは16進数で、0～FFFFFFFFまで)により16進数を10進数に変換することができますが、その逆はありません。そこで2つともBASICで作ってみます。

①10進数→16進数変換

```
300:"C":INPUT N
310:H$=""
320:I= INT (N/16)
330:J=N-16*I
340:H$=CHR$ (J+48
    -(J>9)*7)+H$
350:IF I>0 THEN
    N=I:GOTO 320
360:PRINT H$
370:GOTO 300
```

②16進数→10進数変換

```
400:"D":INPUT H$
410:N=0
420:FOR I=1 TO
    LEN H$
430:P$=MID$ (H$,
    I,1)
440:J=ASC P$-48
450:J=J+(J>9)*7
460:N=N*16+J
470:NEXT I
480:PRINT N
490:GOTO 400
```

①10進→16進変換は、10進→2進変換の方法とほとんど同じです。例題㉖と同じように、10進数を16で割り、商と余りを求めることをくり返せばよいことになります。

例えば、$(246)_D=(F6)_H$ では

商	余り	16進数	アスキーコード
$\frac{246}{16}=15$	J=6	6	54
$\frac{15}{16}=0$	J=15	F	70

答はH$=F6です。

340で(J>9)はJ>9がホントなら−1ウソなら0という答になります。

したがって、J=6ならCHR$(6+48−0×7)=CHR$54、J=15ならCHR$(15+48−−1×7)=CHR$70です。

CHR$↔ASCの意味をよく考えてください。

②16進→10進変換は、2進→10進変換の方法と同じで上の桁から始めればよいでしょう。

たとえば　$(AD3)_H=(2771)_D$では

(A　D　3)$_H$=(2771)$_D$

↓　↓

10×16+13=173

↓　↓

173×16+3 = 2771

プログラムはこの計算を行っています。

440で J=ASC"A"−48=65−48=17

450で J=17+−1×7=10 となります。

以下よくたどってみてください。

例題㉕、㉖、㉗をそのまま打ち込めば、2進数 ⇄ 10進数 ⇄ 16進数 で基本的な変換

（2進数→10進数：GOTO "A" ↵　10進数→2進数：GOTO "B" ↵　10進数→16進数：GOTO "C" ↵　16進数→10進数：GOTO "D" ↵）

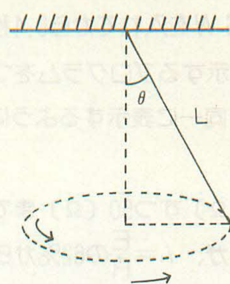
はできますが、ここで少し工夫してもらいたいことがあります。

⑴　演算可能な桁数以上の数を入力した、などのミスを防ぐにはどうすればよいでしょうか。

⑵　2進数⇌10進数変換で、ここでは16ビットまでの1の補数で表していますが、2の補数で表すにはどうすればよいでしょうか。

⑶　表示を見やすいように工夫してください。答の表示では、10進数をSTR$を使って"文字"に直した方がよいでしょう。特に250、350でNがこわれている(Nの数値が変化している)ことに注意してください。

〔例　題〕㉘　75ページの例題①をプログラム化した例題



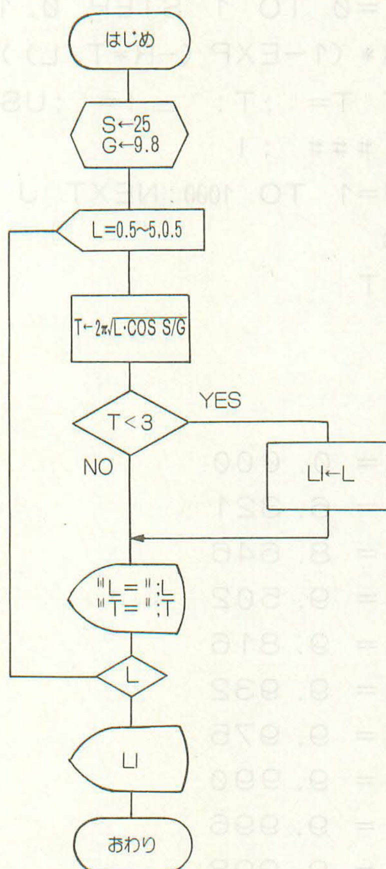
①左図の円すい振り子の周期をT、糸の長さをL、糸の鉛直となす角をθとすると、

$$T=2\pi\sqrt{\frac{L\cdot\cos\theta}{g}}$$

の関係があります。今、糸の長さLを0.5〔m〕から5〔m〕まで、0.5〔m〕刻みで変化させたときのLとTの変化を表示するプログラムをつくりなさい。ただし、g=9.8〔m/sec²〕、θ=25°の場合とします。

②また、それぞれ表示をしたあとで、Tの値が3秒を越えないLの最も大きい値を表示するプログラムをつくりなさい。

■フローチャート



■プログラム例

```
10  REM  フリコ
20  S=25:G=9.8
30  FOR L=0.5 TO 5 STEP 0.5
40  T=2*PI*SQR(L*COS S/G)
50  IF T<3 THEN LI=L
60  PRINT"L=";L;USING"##.
    ##";"␣T=";T
65  FOR J=1 TO 1000:NEXT J
70  USING
80  NEXT L
90  PRINT"T<3---L=";LI;
    "(M)"
100 END
```

■結　果

```
L=0.5  T= 1.35
L=1.   T= 1.91
L=1.5  T= 2.34
L=2.   T= 2.70
L=2.5  T= 3.02
L=3.   T= 3.30
L=3.5  T= 3.57
L=4.   T= 3.82
L=4.5  T= 4.05
L=5.   T= 4.27

T<3---L=2.(M)
```

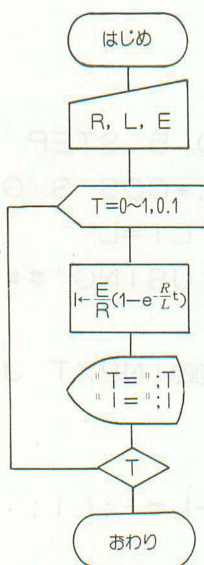
〔例　題〕㉙　　76ページの例題③をプログラム化した例題

SW　E=100〔V〕　L=1〔H〕　i(t)　R=10〔Ω〕

（a）左図の回路において、スイッチＳＷを入れてから0.1秒毎に１秒までの間に流れる電流を表示するプログラムをつくりなさい。ただし、時間と電流を同一に表示するようにしなさい。

（b）次に、抵抗Rを10〔Ω〕から10〔Ω〕ずつ50〔Ω〕まで変化させたとき、回路を流れる電流が、$i=\frac{E}{R}$の62%から65%になるまでの時間を求めるプログラムをつくりなさい。ただし時間は0秒から0.2秒まで0.002秒間隔で調べなさい。

■フローチャート



■プログラム（a）

```
10 REM R-L
20 INPUT"R=";R, "L=";L,
   "E=";E
30 FOR T=0 TO 1 STEP 0.1
40 I=E/R*(1-EXP(-R*T/L))
50 PRINT"T=";T;"␣I=";USIN
   G"##.###";I
55 FOR J=1 TO 1000:NEXT J
60 USING
70 NEXT T
80 END
```

■結　果（a）

```
T=0.    I= 0.000
T=0.1   I= 6.321
T=0.2   I= 8.646
T=0.3   I= 9.502
T=0.4   I= 9.816
T=0.5   I= 9.932
T=0.6   I= 9.975
T=0.7   I= 9.990
T=0.8   I= 9.996
T=0.9   I= 9.998
T=1.    I= 9.999
```

■プログラム（b）

```
10 REM R-L
20 INPUT"L=";L,"E=";E
30 FOR R=10 TO 50 STEP 10
40 FOR T=0 TO 0.2 STEP 0.002
50 I=E/R*(1-EXP (-R*T/L))
60 IF (I>=E*0.62/R)AND(I<=E*0.65/R)
   THEN PRINT"TR(";R;")=";T;"(S)"
   :GOTO 80
70 NEXT T
80 NEXT R
90 END
```

■結　果（b）

```
TR(10.)=0.098(S)
TR(20.)=0.05 (S)
TR(30.)=0.034(S)
TR(40.)=0.026(S)
TR(50.)=0.02 (S)
```

３．変数の種類と使いかた

これまで変数を使ってプログラムを作ってきましたが、ここで変数の種類と使いかたについて説明します。

(1)変数の種類

変数には大きく分けて、数値変数と文字変数の２種類があります。それらは、さらに固定変数、単純変数、配列変数に分類されます。

- 変数
  - 数値変数
    - 数値固定変数……Ａ、Ｂ、Ｃなど
    - 数値単純変数……ＡＢ、Ｂ１、ＣＣなど
    - 数値配列変数……ＡＢ(２，３)など
  - 文字変数
    - 文字固定変数……Ａ＄、Ｚ＄など
    - 文字単純変数……ＢＢ＄、Ｃ２＄など
    - 文字配列変数……ＡＢ＄(２，３)など

【変数の名前】

変数の名前には、Ａ～Ｚのアルファベット１文字か、あるいはＡＡ、ＢＣ、Ａ５などのように文字や数字を組み合わせた２文字が使用できます。

●変数名には次のような文字は使用できません。

①カタカナおよび記号。(ただし、＄記号は文字変数を表す記号として使用されます。)

②予約語。(予約語とは、本機がBASIC命令や関数命令などの名前として使用しているものです。)

〈例〉 ＰＩ、ＩＦ、ＴＯ、ON、ＳＩＮなど

③小文字のアルファベット。(小文字のアルファベットは大文字に変換されます。)

●変数名は必ずアルファベットで始まっていなければなりません。たとえばＡ５は変数名になりますが、５Ａは変数名になりません。

●変数名に３文字以上使用してもエラーにはなりませんが、本機が変数名と判断するのは最初の２文字だけです。

したがって、たとえば変数名にKOTAE1とKOTAE2を使用しますと、本機はこれらを区別できず、ＫＯという同じ変数と判断します。

●固定変数(１文字で指定する変数)では、数値変数と文字変数に同じ名前を使用することはできません。たとえば、数値変数としてＡを使用しているとき、文字変数Ａ＄を使用することはできません。(この場合、後からＡ＄に文字を代入しますと、前の数値変数Ａが消されます。)

【変数の長さ】

変数はその種類によって、長さ(その変数に格納できるデータの最大の長さ)が決められています。

次に各変数の長さを示します。

変数の種類	変数の長さ
数値変数	有効数字10桁まで(仮数部10桁、指数部2桁)
文字固定変数	7文字まで
文字単純変数	16文字まで
文字配列変数	16文字(ただし、1〜255文字まで設定可能)

- それぞれの文字変数の長さを超える文字を記憶させようとした場合、超えた分の文字は無視(切り捨て)されます。

(2)固定変数

これまでの説明やプログラム例で、変数名をA、B、C…またはA＄、B＄、C＄…と指定して使用しましたが、これらの変数を固定変数(1文字変数)と呼びます。

固定変数は、変数として使用されるエリア(区域)がメモリ上に独立して確保されており、このエリアにはプログラムなどが書き込まれることはありません。

このエリアをデータ専用エリアと呼びます。

〈データ専用エリア〉(容量208バイト)

	変数名	データの記憶場所
26	Z または Z＄	
⋮	⋮	⋮ 固定変数(26メモリ)
3	C または C＄	
2	B または B＄	
1	A または A＄	

この図で示すように、このエリアは26に区切られており、変数A、変数B……として使用される場所はすべて決まっています。

また、同じ名前の数値固定変数と文字固定変数(たとえばAとA＄)は同じ場所が使用されます。

(3)単純変数

単純変数は変数名をAAやB1のように2文字(あるいはそれ以上)で指定する変数です。

この変数は固定変数のように使用される場所は決まっておらず、初めて使用したときにメモリ内(プログラム・データエリア内)に自動的に確保されます。

また、同じ変数名でも数値単純変数と文字単純変数は別々に確保されるので、たとえばABとAB＄を同時に使用することもできます。

⑷配列変数(一次元配列、二次元配列)

数学などで、同じ性質の複数個のデータを表すとき、たとえば

X_1, X_2, X_3……X_n

のように、1つの変数名に添字をつけて表す場合があります。

BASICでも同様に、1つの変数名に( )で囲った添字をつけて、次のように表すことができます。

X(1)、X(2)、X(3)……X(10)

このように添字をつけて表す変数を配列変数と呼びます。

配列変数を使うとデータの集計などの作業がしやすくなります。

配列変数を使用するときは、事前にDIM(ディメンジョン)命令によって、配列名とその大きさを定義(宣言)しておかなければなりません。(くわしくは、BASICの各命令の説明を参照してください。)

DIM命令で定義をすることによって、その大きさの配列変数がメモリ(プログラム・データエリア)上に確保されます。

⑴一次元配列

添字が1個だけのものを一次元配列といいます。たとえば、

DIM X(5)

を実行すると、次のように配列名Xの6個の変数(配列要素)が確保されます。

X(0)	X(1)	X(2)	X(3)	X(4)	X(5)

注) この図は配列が横に並んでいるように書いていますが、縦に並んでいると考えても同じです。

実際の使いかたは114ページを参照してください。

⑵二次元配列

本機は添字を2個まで使用できます。

添字が2個のものを二次元配列といいます。

一次元配列が、いうなれば横一列(または縦一列)の配列だったのに対し、二次元配列は縦と横の配列といえます。下の図を見てください。これは

DIM X(3, 5)

と定義したときに確保される変数(配列要素)を表したものです。

X(0,0)	X(0,1)	X(0,2)	X(0,3)	X(0,4)	X(0,5)
X(1,0)	X(1,1)	X(1,2)	X(1,3)	X(1,4)	X(1,5)
X(2,0)	X(2,1)	X(2,2)	X(2,3)	X(2,4)	X(2,5)
X(3,0)	X(3,1)	X(3,2)	X(3,3)	X(3,4)	X(3,5)

これは、ＤＩＭ　Ｘ(５)と定義したときの図と見比べれば、おわかりのように、一次元の配列を４段重ねた形です。

したがって、変数(配列要素)は横に６個(５＋１個)のものが縦に４段(３＋１段)重なっていますので、24個確保されていることになります。

(3)文字配列変数

配列変数にも文字変数があり、配列名に＄(ドル)記号をつけて表します。

```
ＤＩＭ　Ｚ＄(９)              ＤＩＭ　Ｘ１＄(２，１)
ＤＩＭ　Ｙ＄(５，４)                  Ｘ１＄(Ø，１)＝"ヨコハマ"
```

【文字配列変数の拡張】

文字固定変数は１個の変数に最大７文字、文字単純変数は１個の変数に最大16文字記憶できる固定長の変数でしたが、文字配列変数は1〜255文字の範囲で、任意に変数の長さを指定することができます。

ＤＩＭ　Ｃ＄(９)＊３Ø

　　Ｃ＄(Ø)〜Ｃ＄(９)の各変数は、それぞれ最大30文字まで記憶することができます。

ＤＩＭ　Ｎ＄(５，４)＊６

　　Ｎ＄(Ø，Ø)〜Ｎ＄(５，４)の各変数には、それぞれ最大６文字まで記憶することができます。

このように、ＤＩＭ文に＊式をつけて、変数の長さを指定します。

なお、長さを指定しない(＊式がない)ときは自動的に16文字が指定されます。

(4)メモリの構成と変数

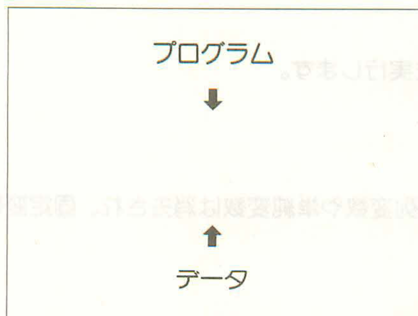
本機はプログラム、データ(固定変数を含む)などを本体内蔵のメモリに記憶します。

プログラム、単純変数、配列変数が記憶される領域をプログラム・データエリアと呼びます。

プログラムはプログラム・データエリアの前方から、データは後方からエリアを確保します。そのためにプログラムの長さによって変数として使える大きさが変わってきますので注意が必要です。

なお、固定変数はデータ専用エリアとして別に確保されています。

〈プログラム・データエリア〉



←プログラムはこちら側から書き込まれていきます。

←Ａ(１)やＢ＄(５，８)のような配列変数およびＡＡやＢ１＄のような単純変数はこちら側から確保されます。

次に、各変数を確保する場合に使用するバイト数およびプログラムの各命令などの占めるバイト数を記しておきますので参考にしてください。

● 変数の場合

変　数	変数の名前	データ
数値単純変数 数値配列変数	7バイト	8バイト
文字単純変数	7バイト	16バイト
文字配列変数	7バイト	指定されたバイト数

※たとえば、DIM　B$(2, 3)*1Øと指定した場合は、次のバイト数を使用します。
7バイト(変数名)+10バイト(文字数)×12個=127バイト　になります。

● 命令の場合

構成要素	ラインナンバー	命令文、関数	↵、その他
使用バイト数	3バイト	2バイト	1バイト

※たとえば、下のラインを入力した場合、次のバイト数を使用します。

1Ø　PRINT　A；"X—"↵

3バイト(ラインナンバー)+2バイト(命令文)+1バイト(↵)+6バイト(その他1×6)=12バイト

●プログラム・データエリアの残りのバイト数はFRE命令で求めることができます。(くわしくはFRE命令を参照してください。)

(5)変数をクリア(消去)するには

(1)個々の変数の内容を消去するには、次のようにします。

〈例〉 A=Ø
　　B(1)=Ø　}　数値変数はØを代入することにより消去します。
　　A$=""
　　B1$=""　}　文字変数は""を代入することにより消去します。""を代入するということは文字変数を文字が入っていない状態にするということです。この状態をNull(ヌル)状態と呼びます。

(2)CLEAR命令

すべての変数を一度に消去するには、CLEAR命令を実行します。

CLEAR命令の一般形は次のようになります。

　CLEAR

この命令を実行すると、メモリ内に確保されていた配列変数や単純変数は消去され、固定変数の内容も消去されます。

● この命令ではプログラムは消去されません。

4. デバッグ

プログラムを実行した場合に、何かの誤りで迷走したり思わぬ結果が出たりすることがあります。このような場合、プログラムのリストを調べて誤りを探しますが、それだけではなかなか見つけにくいことがあります。

このようなとき、プログラムを1行ずつ実行させ、その経過をたどりながらプログラムの誤りを探していけば見つけやすくなります。

ここでは、1行ずつ経過をたどりながらプログラムを実行する方法について説明します。(このような方法をトレースといいます。また、プログラムの誤りを探し、修正することをデバッグ(虫取り)といいます。)

(1)デバッグのしかた

①トレースによるデバッグはRUNモードで行いますので、RUNモードにしてください。

②TRON [↵] と押して、トレースモードにします。

③RUN [↵] と押してプログラムの実行を開始します。(最初の行の実行が終われば、実行した行番号を表示して実行が停止します。)

④その後は [↓] キーを押します。(1行だけ実行して停止します。)ただし、INPUT命令でのデータ入力は、通常のプログラム実行と同じように [↵] キーで行います。

⑤プログラムの実行順序の確認や、各行(各ライン)実行後の変数の内容確認などを行いながらトレースを進め、プログラムが正しく実行されているかどうかチェックします。
もし、正しく実行されないときは、その原因を探して修正します。

⑥デバッグが終了すれば、TROFF [↵] と押してトレースモードを解除します。

次にごく簡単な例を示します。

〈例〉
```
10  INPUT "A=";A, "B=";B
20  C=A*2
30  D=B*3
40  PRINT "C=";C;" D=";D
50  END
```

〈実行〉 RUNモード

TRON [↵]		
RUN [↵]	A=_	INPUT命令実行
8 [↵] (データ入力)	B=_	
9 [↵] (データ入力)	10:	←10行終了
[↓]	20:	←20行終了
[↓]	30:	←30行終了
[↓]	C=16. D=27.	←PRINT命令実行
	40:	←40行終了
[↓]	>	実行終了 (プロンプト表示)

トレース中で行番号を表示しているときは、マニュアル操作で変数の内容を呼び出し、予定した値になっているかどうかチェックできます。

また、このとき [↑] キーを押せば、押している間、止まっている行の内容を表示します。(なお、[↑] キーを押して離したときはプロンプト記号(>)が表示されますが、[↓] キーで続けて実行できます。)

- トレース中に、LOCATE命令で指定された画面の位置に結果などが表示された場合、次の行番号は、結果などが表示された行の次の行から表示されます。(LOCATEについては、BASICの各命令の説明を参照)
- LOCATE命令により表示開始位置が指定されているときに、マニュアル操作で変数の呼び出しや計算などを行いますと、表示開始位置の指定は解除されます。

(注)トレースモードは、TROFF [↵] と押すか、[SHIFT] + [CA] (または [2ndF] [CA])と押す、または電源が切れるまで設定状態が保持されます。

(2)プログラムの途中で実行を停止させてチェックする場合

プログラムの実行を停止させたい位置にSTOP命令を書いておけば、STOP命令を実行した時点でブレークメッセージ(BREAK IN 行番号)を表示して、プログラムの実行が停止します。

このとき

①マニュアル操作で変数の内容をチェックする。

②続いて [↓] キーの操作で、以降の行を1行ずつトレースする。

などの操作でデバッグを行います。その後、通常の実行状態にもどすときは

CONT [↵]

と押します。

- 通常のプログラム実行中に [BREAK ON] キーを押すと、現在実行している行の終わりで実行を停止し、ブレークメッセージを表示します。このときも前記と同様の操作を行うことができます。
  なお、ブレーク状態(停止状態)のときに [↑] キーを押せば、押している間停止しているプログラム行が表示されます。
- トレースモードを設定しない場合でも、[BREAK ON] キーやSTOP命令などにより一時停止しているプログラムを [↓] キーにより1行ずつ実行させていくことができます。この場合も表示はトレースモードが設定されている場合と同じようになります。
  なお、[↓] キーを押したままにしますと、連続的にトレースが実行されます。

＝ご注意＝

トレースモードにおいて、注釈行* を実行した場合は、その行の行番号は表示されません。

この場合は、それ以前に実行した注釈行以外の行の行番号が表示されます。

※注釈行とは、行番号に続いて'(シングルクォーテーション)が書かれている行です。

5. プログラムのファイル

本機はプログラムをファイルとして本機の内部メモリに保存しておくことができます。
本機内部には、プログラムやデータを記憶するメモリがあり、初期状態(何も使用していない状態)のときは、プログラム・データエリアになっています。このときの大きさ(容量)は30435バイトです。
このメモリの一部をプログラム・データエリアと切り離して、ファイルエリア(ラムファイルエリア)として確保し、このエリアにプログラムを保存します。
本機で、実行できるプログラムはプログラム・データエリアに入っているプログラムですが、ファイルしたプログラムを実行したいときは、ファイルエリアからプログラム・データエリアに呼び出して、実行します。
ここでは、BASICプログラムのファイルのしかたについて説明します。
(注)BASICプログラムだけでなく、テキストのファイルもあります。TEXTモードの説明を参照してください。

(1)ファイルに関する命令

次に、ファイルに関する命令を簡単に述べます。くわしくは、BASICの各命令の説明を参照してください。

〈命　令〉	〈機　能〉
FILES	登録されているファイルのファイル名を表示します。
LFILES	登録されているファイルのファイル名をプリンタで印字します。
KILL	ファイルを消去します。
LOAD	BASICプログラムをファイルエリアからプログラム・データエリアに呼び出します。
SAVE	プログラム・データエリア内のBASICプログラムをファイルエリアに保存(登録)します。

●ファイル名
KILL、LOAD、SAVE命令を実行するときはファイル名を指定する必要があります。ファイル名はファイルの見出しのようなものです。
ファイル名は最大8文字で構成され、次の文字が使用できます。

A～Z、a～z、0～9、#、$、%、&、'、(、)、ー、@、{、}、カタカナ

また、ファイル名には拡張子を付けることができます。拡張子はファイルの種類を区別するためなどに利用します。
拡張子はピリオドと3文字以内の文字で構成され、ファイル名の直後に付けます。使用できる文字は、ファイル名と同じものです。
なお、SAVEやLOAD命令でBASICプログラムをセーブしたりロードしたりするとき、またKILL命令でファイルを消去するとき、拡張子を省略すると自動的に「.BAS」が指定されます。

ファイル名の完全な記述は次の形式になります。

"ファイル名.拡張子"

(2)プログラムの登録(保存)(SAVE命令)

プログラム作成後、そのプログラムを登録する場合は、RUNモードまたはPROモードで次のような操作を行います。

〈例〉 SAVE "TEST" ↵
　　　　　　↑ファイル名

ファイルとして登録する場合、必ずファイル名が必要です。ファイル名には、拡張子を付けることができます。(ファイル名の説明を参照)

この例では、拡張子を省略していますが、この場合、".BAS"が自動的に付けられます。

なお、ファイルエリアはSAVE命令でプログラムを登録したとき、自動的に必要な大きさが確保されます。ただし、メモリの残りが少なく、必要な大きさが確保できないときはエラー60になります。

(3)ファイルの登録の確認(FILES、LFILES命令)

プログラムを登録した場合、FILESまたはLFILES命令で確認することができます。

〈例〉 FILES ↵

と操作すれば、登録されているファイル名などが画面に表示されます。

登録されているファイルが多い場合は ↓ キーを押していくことにより、画面に呼び出すことができます。戻すときは ↑ キーを押します。

〈例〉 LFILES ↵

と操作をすれば、登録されているファイルのファイル名などが印字されます。(ただし、別売のプリンタCE-126Pが接続されているときのみ有効です。)

(4)プログラムの呼び出し(LOAD命令)

登録したプログラムは次のような操作で呼び出すことができます。

〈例〉 LOAD "TEST" ↵

この場合、ファイル名が"TEST.BAS"のファイル内容(プログラム)が呼び出されます。

なお、登録したときに拡張子を付けた場合は、拡張子まで完全に入力してください。

〈例〉 FILES ↵　　ファイル名を表示させます。

↓ … ↓
↑ … ↑　　画面のファイル名の前にある矢印(➡)を呼び出したいファイル名の前へ移します。

SHIFT + M (LOAD)　　矢印(➡)で示したファイルの内容(プログラム)が、プログラム・データエリアに呼び出されます。

このようにFILES命令で呼び出した、ファイル名を選んで、プログラムを呼び出すこともできます。

(5)記録されているファイルの消去(KILL命令)

1つのファイルを消去するときは、次のような操作を行ってください。

〈例〉 KILL "TEST. BAS" ↵

この場合、"TEST. BAS"というファイルが消去されます。

KILL命令では、記録されているファイル名の通りに、ファイル名を拡張子まで入れてください。ただし、拡張子が". BAS"の場合は拡張子を省略できます。

なお、TEXTモードで消去する方法もあります。(第5章を参照)

6. テープへの記録、読み込み

別売のプリンタ/カセットインターフェイスCE-126Pとテープレコーダ(CE-152)があれば、プログラムやデータをテープに記録したり、読み込んだりできます。ここでは、プログラムをテープに記録する、あるいはテープから読み込む場合の操作方法について説明します。

(1)カセットに関する命令

次にカセットに関する命令を簡単に述べます。くわしくは個々の命令の説明を参照してください。

〈命　令〉	〈機　能〉
CLOAD	テープに記録されているプログラムを読み込みます。
CLOAD ?	テープに記録されているプログラムと計算機内のプログラムの照合を行います。
CLOAD M	テープに記録されている機械語プログラムを読み込みます。
CSAVE	計算機内のBASICプログラムをテープに記録します。
CSAVE M	計算機内の機械語プログラムをテープに記録します。
INPUT #	テープに記録されているデータを変数に読み込みます。
PRINT #	計算機内のデータをテープに記録します。

(2)準備

本機、CE-126P、テープレコーダ(CE-152)を用意し、それぞれを接続してください。接続方法はCE-126PおよびCE-152の取扱説明書を参照してください。

なお、CE-126Pは本機の左の周辺機器接続端子(11ピン)❷に接続してください。

〈おねがい〉

カセットテープに、プログラムおよびデータを記録するときに、リモート機能をご使用になる場合は、必ず次の操作を行ってください。

(操作手順) ① BASIC

② POKE &H78DD, 1ØØ ↵

③ POKE &H78E2, 7 ↵

この操作は、リセットスイッチ❺を押したり、電池交換をしない限り、再設定の必要はありませんが、リセットスイッチ❺を押したり、電池交換をしたときは再度行ってください。

〈テープレコーダに関するご注意〉

- テープから読み込み、または照合を行うときは記録を行ったテープレコーダを使用してください。
- テープレコーダのヘッドの清掃を行ってください。ヘッドなどが汚れていますと、ひずみの増加、レベルの低下などが起こります。
- テープは一般に市販のものをご使用ください。ただし、極端に周波数特性の悪いものや、傷や折り目があるものは使用しないでください。
- 一度使用したテープに新しくプログラムなどを書き込む場合は、書き始める前を少し消去してから記録命令を実行してください。

(3)テープへの記録方法

①本機にプログラムを入れてください。

②CE-126PのリモートスイッチをOFF位置にしてください。

③テープをテープレコーダに入れ、記録を始める位置までテープを送ってください。

④CE-126PのリモートスイッチをON位置にしてください。

⑤テープレコーダの記録ボタンを押してください。(このとき、テープはスタートしません。)

⑥記録命令を入力し、実行してください。テープがスタートします。

〈例〉 RUNモード(またはPROモード)を指定

CSAVE "PRO-1" ↵

記録命令を実行すると、最初にしばらくピー音が記録され、その後に ビビビビ というような音でプログラムが記録されます。

⑦記録が終了しますと、プロンプト記号が表示され、テープが止まります。テープを巻き戻して次項の方法で“照合”を行ってください。

- テープの早送り、巻き戻しなどを行うときは、CE-126PのリモートスイッチをOFF位置にしてください。
- プログラム中で、データの記録(PRINT#命令の実行)、または読み出し(INPUT#の実行)を行うときは、プログラム実行前にテープレコーダの操作をしておいてください。

記録を行うとき、記録を始めたときのテープカウンタの数値を紙などに控えておけば、記録位置や、記録されていない位置を探し出すときに便利です。

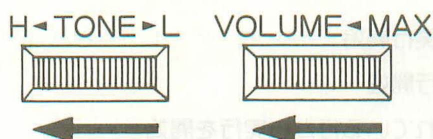
(4)計算機内とテープのプログラムの照合

プログラムをテープに記録したあと、まちがいなく記録されたかどうか照合して確認します。

①CE-126Pのリモートスイッチを0FF位置にしてください。

②テープをテープレコーダに入れ、照合する内容が記録されている位置の直前までテープを進めてください。

③音量調整つまみ、音質調整つまみを図の矢印方向いっぱいに回しておいてください。



(注)つまみの位置を少し調整したほうが良好な場合があります。

④CE-126PのリモートスイッチをON位置にしてください。

⑤テープレコーダの再生ボタンを押してください。(このとき、テープはスタートしません。)

⑥照合命令を入力し、実行してください。テープがスタートします。

〈例〉 RUNモード(またはPROモード)を指定

CLOAD?"PRO−1" ↵

● プログラムが見つかり、照合が始まると画面の下の行の右端桁に＊マークが表示されます。

⑦両方の内容がすべて一致していれば実行を終了し、プロンプト記号が表示されます。

もし、エラー82になった場合は、もう一度最初から照合を行ってください。

それでもなお、エラーになる場合は、もう一度"記録"から行ってください。

(5)テープからの読み込み

プログラムをテープから計算機に読み込む場合は、照合命令を読み込み命令にかえて、照合の場合と同じ手順で行ってください。

読み込み命令:CLOAD

〈例〉 CLOAD"PRO−1" ↵

もし、読み込み途中でエラー80になった場合は、もう一度最初から読み込みを行ってください。

7. プログラムの実行開始方法とラベルについて

(1)プログラムの実行開始方法

プログラムの実行開始方法には次の方法があります。

(1)RUN命令によるもの

RUN ↵ ………………プログラムの先頭から実行開始

RUN　行番号 ↵ ……指定した行番号から実行開始

RUN　ラベル ↵ ……指定したラベルが書かれている行から実行を開始

(2)GOTO命令によるもの

GOTO　行番号 ↵ …指定した行番号から実行開始

GOTO　ラベル ↵ …指定したラベルが書かれている行から実行を開始

ラベルは、" "（ダブルクォーテーション）で前後を囲うか、*（アスタリスク）を前につけて指定します。（くわしくは、次ページを参照）

〈例〉 RUN"AB"
GOTO*AB

これらの開始方法により、状態の解除や変数の消去などに違いがありますので次に示します。

RUN命令による実行	GOTO命令による実行
●ウエイト(WAIT)はØに設定されます。	●ウエイト(WAIT)指定は保持されます。
●表示フォーマット(USING)指定は解除されます。	●表示フォーマット(USING)指定は保持されます。
●トレースモードの設定(TRON)は保持されます。	●トレースモードの設定(TRON)は保持されます。
●配列変数(DIM指定)、単純変数は消去されます。	●配列変数(DIM指定)、単純変数は保持されます。
●固定変数の内容は保持されます。	●固定変数の内容は保持されます。
●READ文に対するDATAの初期化を行います。	●READ文に対するDATAの初期化は行われません。
●PRINT=LPRINT指定を解除します。	●PRINT=LPRINT指定は保持されます。
●パラレルポートを閉じます。	●パラレルポートを閉じません。

注）プログラムをRUN命令で実行したとき、配列変数や単純変数は消去されます。データを残しておきたいときは、GOTO命令で実行を開始してください。

(2)ラベルについて

次の例のように、ラベルはプログラムの行の先頭(行番号の次)に見出しとして書いておくものです。GOTOやGOSUB、THENなどのジャンプ先としてラベルを指定すれば、実行時、指定したラベルを探して、そこへジャンプします。

```
〈例〉 1Ø    "AB" : INPUT  …
       ⋮         ↖ラベルAB
      5Ø    *CDE : CLEAR  …
       ⋮         ↖ラベルCDE
      1ØØ    IF  …  THEN"AB"  ←ラベルABへジャンプ
       ⋮
      15Ø  GOTO  *CDE  ←ラベルCDEへジャンプ
```

また、プログラム実行開始命令であるRUNやGOTO命令で指定すれば、指定したラベルのある行からプログラムを実行することができます。たとえば、使用できる行番号の範囲内で複数のプログラムの先頭にラベルを付けて本機に書き込んでおき、ラベルを指定して実行を開始すれば、必要なプログラムを実行させることができます。

(1)" "で囲んだラベルおよびラベル指定で使用できる文字

" "で囲んだラベルには、英字、数字、カタカナ、記号などが使用できます。

```
〈例〉 "ABCDE"
      "X1Ø"
      "サブ ルーチン"
```

(2)*を用いるラベルおよびラベル指定で使用できる文字

*記号を先頭につけるラベルには、英字か英字と、それに続く数字が使用できます。

```
〈例〉 *START
      *S123
```

- ラベルは必ず英文字で始まっていなければなりません。
- 英字の小文字は大文字に変換されます。
- 予約語のスペルで始まる文字列は使用できません。
- カタカナや記号は使用できません。

本機では" "で囲んだラベルと、*を用いるラベルを混合して使うことができますが、1つのプログラムの中では、どちらか一方に統一して使用されることをお勧めします。
なお、*を用いるラベルは多くのパソコンでも用いられています。

第5章　TEXTモード（テキストエディタ）

ＴＥＸＴモード(テキストエディタ)では、アスキー形式でのプログラムの入力、編集、ＳＩＯへの入出力などが行えます。

本機のＢＡＳＩＣの各命令は、中間コードと呼ばれる2バイトコードに変換して記憶しています。このコードはハードウェアやＢＡＳＩＣインタプリタにより異なるため、そのままではパソコン等との通信はできません。

アスキーコードは、アルファベットや数字、基本的な記号がほとんどの機種で共通であるため、パソコン等ではアスキーコードでの通信が多く行われています。

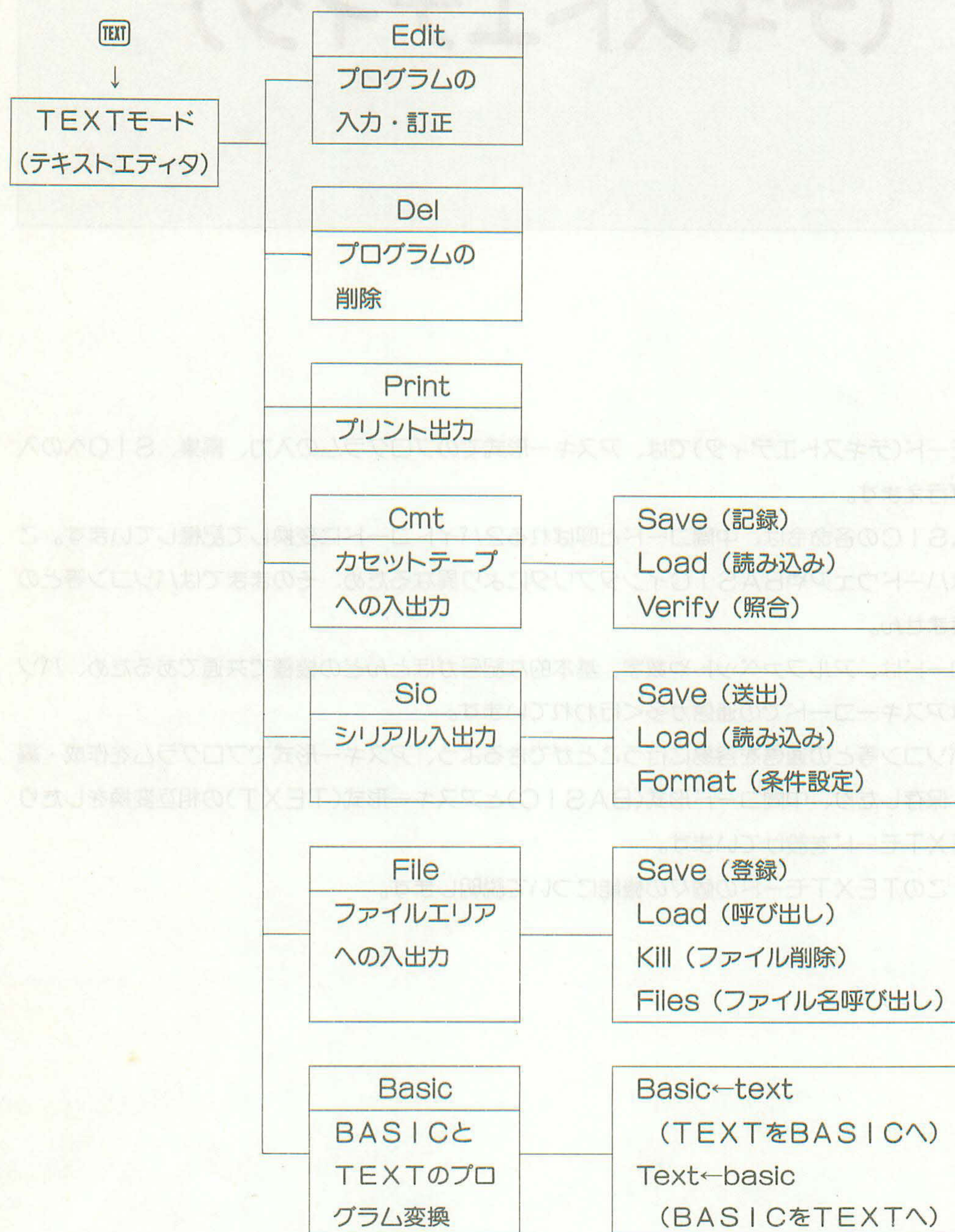
本機ではパソコン等との通信を容易に行うことができるよう、アスキー形式でプログラムを作成・編集したり、保存したり、中間コード形式(ＢＡＳＩＣ)とアスキー形式(ＴＥＸＴ)の相互変換をしたりできるＴＥＸＴモードを設けています。

本章では、このＴＥＸＴモードの個々の機能について説明します。

1. TEXTモード機能一覧

TEXTモードの機能概略図を下記します。参考にしてください。

なお、CASLのソースプログラムの入力・編集も、TEXTモードで行います。



2. TEXTモードの使いかた

(1)TEXTモードの設定

RUNモードやPROモードなどで TEXT キーを押すと、右の画面になります。この画面をメインメニュー画面と呼びます。

```
*** TEXT EDITOR ***

Edit   Del    Print Cmt
Sio    File   Basic
```

この画面で、表示されている各機能をその頭文字(大文字)に相当するキーを押して選びます。

各機能を選ぶと、それぞれの機能のメニュー画面になったり、機能が働いたりします。

- 各機能が働いている状態で、動作を止めたり、メニュー画面に戻したり、メインメニュー画面に戻したりするときは、 BREAK キーを押してください。
  ただし、エラー状態を解除したり、ファイル名など入力中の文字などを消去するときは CLS キーを押してください。
- TEXTモードは、 BASIC 、 SHIFT ＋ CASL キーの操作および電源を切ることにより解除されます。

(2)エディット機能（Edit）

メインメニュー画面で E キーを押せば、エディット機能が選ばれ、エディット画面になります。

E

```
TEXT EDITOR
<
```

- エディット機能では、プロンプト記号が“<”になります。（BASICモードでは“>”）
- TEXTプログラムを書き込むときは、BASICプログラムの場合と同じように、行番号を先頭につけて書き込みます。ただし、BASICのように、自動的に行番号の後ろにコロン(：)を付けたり、命令の後ろにスペースを入れたりはしません。入力したとおりに書き込まれます。
- 行の順番は、行番号の順番に並べ替えられます。
- 行番号は、1～65279の範囲で付けることができます。この範囲を超えている場合、または行番号がない場合はエラー(LINE NO. ERROR)になります。エラーは CLS キーで解除してください。
- メインメニュー画面に戻るときは BREAK キーを押してください。

注）行番号の次が数字で始まるようなテキスト行を入力することはできません。
数字で始まるような行を入力するときは '(シングルクォーテーション)で行番号と数字を区切って入力してください。
〈例〉 5Ø'1ØØ FORMAT(17X, A) ↵
行番号 ↑シングルクォーテーション

〈例〉次のプログラムを入力します。

```
1ØINPUT  A
2ØB=A*A
3ØPRINT  A, B
4ØEND
```

```
1ØINPUT[SPACE]A[↵]        1ØINPUT  A
2ØB=A*A[↵]               2ØB=A*A
3ØPRINT[SPACE]A,B[↵]      3ØPRINT  A, B
4ØEND[↵]                 4ØEND
```

プログラムの編集

TEXTプログラムの変更、修正などはBASICプログラムの変更、修正と同じ方法で行うことができますので参照してください。

なお、BASICのLIST命令、RENUM命令に代わる命令として、L命令(リスト)、R命令(リナンバー)が用意されています。働きはLIST命令、RENUM命令と同じですので参照してください。

ただし、TEXTモードでのリナンバーは、行頭の番号だけを付け直します。

BASICプログラムをTEXTプログラムに変換して、リナンバーを行いますとGOTOやTHEN、GOSUB、RESTORE命令などの後の行番号は変更されていませんので、再びBASICプログラムに変換したとき、正しく動作しなくなりますので注意してください。

L命令書式　(1)　L [↵]

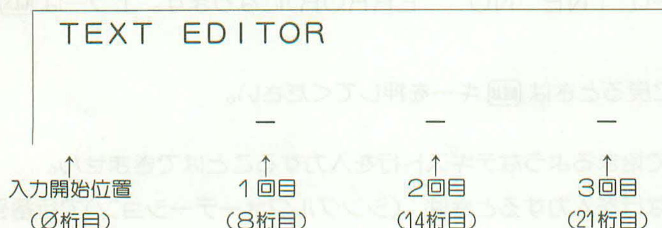
　　　　　　(2)　L行番号 [↵]

　　　　　　(3)　Lラベル [↵]

R命令書式　　R[新行番号][,[開始行][,増分]] [↵]

[TAB] キーの働き

エディット機能の中での [TAB] キーは、次の図に示すようにカーソルを送ります。



最初 [TAB] キーを押すとカーソルが8桁進み、2回目を押すと6桁進みます。3回目以降は7桁ずつ進みます。

(3)TEXTプログラムの消去(Delete)

メインメニュー画面で D キーを押せば、消去確認画面になります。

D

```
    *** TEXT EDITOR ***

TEXT DELETE OK? (Y)
```

Y キーを押せば、TEXTプログラムなど、テキストエリアの内容が消去され、メインメニュー画面に戻ります。

Y 以外のキーを押すと、消去をせずにメインメニュー画面に戻ります。

- TEXTモードでの記憶内容がない場合は、メインメニュー画面で D キーを押しても何も行いません。

(4)TEXTプログラムの印字(Print)

別売のプリンタCE-126Pを接続して電源を入れ、メインメニュー画面で P キーを押せば印字を開始します。

P

```
    *** TEXT EDITOR ***

--- PRINTING ---
```

印字が終われば、メインメニュー画面に戻ります。

- 印字を途中で止めるときは BREAK キーを押してください。
- プリンタの電源が入っていないときや、プリンタが接続されていないときは、メインメニュー画面で P キーを押しても何も行いません。

(5)カセットへの記録、読み込み、照合(Cmt)

メインメニュー画面で C キーを押せば、カセットのメニュー画面になります。

C

```
      << CMT >>

  Save  Load  Verify
```

このメニュー画面で、記録(Save)、読み込み(Load)、照合(Verify)を選びます。
それぞれの頭文字に相当するキーを押してください。

- 各機能を実行する前に、別売のプリンタCE-126P、カセットテープレコーダCE-152を接続し、記録(録音)または読み込み(再生)の準備をしておいてください。

①カセットへの記録(Savc)

カセットのメニュー画面で S キーを押せば、ファイル名の入力待ちになります。

S

```
          << CMT >>

➡Save   Load   Verify
FILE NAME=?
```

ファイル名を入力して ↵ キーを押せば、記録が開始されます。

〈例〉ファイル名を"TEXT"とします。

TEXT

```
          << CMT >>

➡Save   Load   Verify
FILE NAME=TEXT_
```

↵

```
          << CMT >>

    ---  SAVING  ---
```

記録が終了するとカセットのメニュー画面に戻ります。正しく記録されたか確認するため、照合(Verify)を行ってください。

- ファイル名の入力待ちのとき、ファイル名を入力せずに ↵ キーを押せば、ファイル名なしでカセットテープに記録します。
- ファイル名は8文字以下の文字で指定してください。
- TEXTモードでの記憶内容がない場合は、↵ キーを押すと何もせずにカセットのメニュー画面に戻ります。

【ファイルエリアの内容を一括してテープに記録する方法】

上記の記録操作で、ファイル名を"＊．＊"とすればファイルエリアに登録されている内容(プログラムファイル)を一括してテープに記録することができます。

この内容をテープから読み込む場合は、次項の読み込みの操作でファイル名を"＊．＊"として読み込みます。

②カセットからの読み込み

カセットのメニュー画面で L キーを押せば、ファイル名の入力待ちになります。

L

```
          << CMT >>

 Save  ➡Load   Verify
FILE NAME=?
```

ファイル名を入力して ↵ キーを押せば、読み込みが開始されます。

〈例〉ファイル名を"TEXT"とします。

TEXT

```
               << CMT >>

   Save ➡Load   Verify
FILE NAME=TEXT_
```

```
               << CMT >>

     ――― LOADING ―――
```

指定したファイル名(TEXT)を検索し、そのファイル名のアスキー形式で記録された内容を読み込みます。(テキストエリアに読み込みます。)

読み込んでいるときは、画面右下に＊マークが点灯します。

読み込みが終了すると＊マークが消え、カセットのメニュー画面に戻ります。正しく読み込まれたことを確認するため、照合(Verify)を行ってください。

- ファイル名の入力待ちのとき、ファイル名を入力せずに ↵ キーを押すと、テープが回り始めて、最初に出てきたアスキー形式で記録されている内容を読み込みます。
- 指定したファイル名が検索できなかった場合は、テープが回り終わっても検索を続けます。この場合は BREAK キーを押して検索を止めてください。
- 読み込み中にエラーが発生したとき、または BREAK キーで読み込みを中断したときは、正しく読み込みが行われた行までが有効になります。

③カセットとの照合(Verify)

照合は、計算機内のプログラムがカセットテープに正しく記録されたかどうか、あるいは正しく読み込まれたかどうか確認するために行います。

カセットのメニュー画面で V キーを押せばVerifyが選ばれ、ファイル名の入力待ちになります。

```
               << CMT >>

   Save   Load ➡Verify
FILE NAME=?
```

ファイル名を入力して ↵ キーを押せば、照合が開始されます。

〈例〉ファイル名を"TEXT"とします。

TEXT

```
               << CMT >>

   Save   Load ➡Verify
FILE NAME=TEXT_
```

```
<< CMT >>

---- VERIFYING ---
```

指定したファイル名(TEXT)を検索し、そのファイル名のアスキー形式で記録された内容と計算機のテキストエリアの内容とを照合します。

照合中は、画面右下に＊マークが点灯します。

照合が終了すると＊マークが消え、カセットのメニュー画面に戻ります。

もし、内容に不一致があったときは、エラー(VERIFY　ERROR)になります。

●ファイル名の入力待ちのとき、ファイル名を入力せずに ↵ キーを押すと、テープが回り始めて、最初に出てきたアスキー形式で記録されている内容との照合を行います。

●指定したファイル名が検索できなかった場合は、テープが回り終わっても検索を続けます。この場合は BREAK キーを押して検索を止めてください。

(6)シリアル入出力(Sio)

メインメニュー画面で S キーを押せば、シリアル入出力のメニュー画面になります。

S

```
<< SIO >>

Save   Load   Format
```

この画面で、出力(Save)、読み込み(Load)、条件設定(Format)を選びます。それぞれの機能の頭文字に相当するキーを押してください。

①入出力の条件設定(Format)

通信を行う場合の入出力の条件を設定します。入出力の条件は通信する相手側と合わせておく必要があります。

シリアル入出力のメニュー画面で F キーを押すと、まず説明画面になります。しばらく待つか、どれかキーを押すと、条件設定画面になります。

F

```
<< SIO >>
Select   ←, →, ↑, ↓  key
Set      ↵  key
--- push any key ---
```

しばらく待つか、どれかキーを押すと右の画面になります。

```
➡baud rate  =1200
 data bit   =7
 stop bit   =1
 parity     =none
```

➡マークは変更できる項目を示します。↓ 、↑ キーで移動させて変更したい項目を選択します。条件は7種類あり、↓ キーを押していけば画面に出てきます。

```
  parity       =none
  end of line  =CR
  end of file  =1A
➡line number  =yes
```

項目を選択し、▶キーまたは◀キーで条件を切り替えます。ただし、end of file(エンド オブ ファイル)は直接コードを入力します。

そして↵キーを押せば設定されます。↵キーを押さないと、前の条件のままになります。

【条件の説明】

● 通信速度 : 300、600、1200、2400、4800
(baud rate) データ転送速度の指定で、本機では300bps、600bps、1200bps、2400bps、4800bpsの指定ができます。
bps : bit per second (1bps=1ビット/秒)

● ワード長 : 7、8
(data bit) 1文字を何ビット構成で送受信するかを指定します。
本機では7ビットまたは8ビットの指定ができます。

● ストップビット数 : 1、2
(stop bit) ストップビット数を1ビットにするか2ビットにするかを指定します。

● パリティ : none、even、odd
(parity) パリティビットをどのように扱うかを指定します。
none…パリティビットの送受信を行いません。
even…偶数パリティが指定されます。
odd …奇数パリティが指定されます。

● 区切りコード : CR、LF、CR LF
(end of line) プログラムライン(行)の終了を示す区切りコードを指定します。
CR……CR(キャリッジリターン)コードが指定されます。
LF……LF(ラインフィード)コードが指定されます。
CR LF…CRコード+LFコードが指定されます。

● テキスト終了コード : 00〜FF(2桁の16進数値)
(end of file) プログラムなどの終了を示すテキスト終了コードを指定します。

● 行番号付き入出力 : yes、no
(line number)
● TEXTプログラムを出力するときに、行番号を付けて出力するか、付けずに出力するかを指定します。
yes……行番号を付けて出力します。
no ……行番号を付けずに出力します。
● TEXTプログラムを読み込むとき、読み込んだプログラムに行番号(10から10きざみ)を付けるか、付けないかを指定します。
yes……行番号を付けません。
読み込むプログラムに行番号が付いているときに指定します。
no ……自動的に行番号を付けます。
yesを指定しているときに、読み込んだプログラムに行番号が付いていない場合は、エラー(LINE NO. ERROR)になります。

本機の入出力条件は、初期の状態では右のように設定されています。

項　　目	条　件
boud rate	1200
data bit	7
stop bit	1
parity	none
end of line	CR
end of file	1A
line number	yes

入出力条件は前に述べた方法で変更できます。
変更した入出力条件はリセットスイッチ❺を押して、メモリを消去するか、電池交換を行うか、条件の変更を行うまで保持されます。

②出力(Save)

シリアル入出力のメニュー画面で、[S]キーを押すとシリアル入出力端子への出力が開始されます。

[S]

```
<< SIO >>

--- SENDING ---
```

出力が終了すれば、シリアル入出力のメニュー画面に戻ります。

- 出力を中断するときは[BREAK]キーを押してください。メニュー画面に戻ります。
- TEXTモードでの記憶内容がない場合は、[S]キーを押しても何も行われません。

③読み込み(Load)

シリアル入出力のメニュー画面で、[L]キーを押すとシリアル入出力端子へ送られてくるデータの読み込みが開始されます。

[L]

```
<< SIO >>

--- RECEIVING ---
```

読み込みが終了(テキスト終了コードを受信)すれば、シリアル入出力のメニュー画面に戻ります。

- 読み込みを中断するときは[BREAK]キーを押してください。メニュー画面に戻ります。
- データが正常に読み込めない場合や、パリティチェックで異常が発生した場合などにはエラー(I／O　DEVICE　ERROR)になります。このときは[CLS]キーでエラーを解除してください。

⑺プログラムファイル(File)

メインメニュー画面で[F]キーを押すと、プログラムファイルのメニュー画面になります。

[F]

```
        << PROGRAM FILE >>

 Save    Load    Kill    Files
```

この画面で、登録(Save)、呼び出し(Load)、削除(Kill)およびファイル名の確認（Files）を選びます。それぞれの頭文字に相当するキーを押してください。

①TEXTプログラムの登録(Save)

プログラムファイルのメニュー画面で[S]キーを押すと、ファイル名の入力待ちになります。

[S]

```
        << PROGRAM FILE >>

➡Save    Load    Kill    Files
FILE NAME=?
```

ファイル名を入力して[↵]キーを押せば、登録が行われます。

〈例〉ファイル名を"TEST"とします。

TEST

```
        << PROGRAM FILE >>

➡Save    Load    Kill    Files
FILE NAME=TEST_
```

[↵]

```
        << PROGRAM FILE >>

 Save    Load    Kill    Files
```

登録が行われ、メニュー画面に戻ります。

- ファイル名を省略することはできません。ファイル名を入れずに[↵]キーを押すとエラー（ILLEGAL FILE NAME)になります。
- ファイル名は8文字以下の名前と、3文字以下の拡張子を指定することができます。拡張子を付けなかった場合は、自動的に".TXT"が付けられます。
- テキストエリアにTEXTプログラムがない場合は、登録は行われません。

②TEXTプログラムの呼び出し(Load)

プログラムファイルのメニュー画面で L キーを押すと、登録されているファイル名を表示し、最初のファイル名の左側に"LOAD ➡"と表示されます。(何も登録されていないときは、画面は変わりません。)

L
(画面は、すでにこれらのプログラムが登録されている場合の例です。)

```
LOAD ➡ABC      .TXT
      PRO      .TXT
      サンプ ルØØ1.BAS
      TEST     .TXT
```

↓、↑ キーで"LOAD ➡"表示を呼び出したいファイル名の前に移し、↵ キーを押せば、そのファイルの内容が呼び出され、メニュー画面に戻ります。

- TEXTモードで呼び出すことができるプログラムは、TEXTモードで登録したプログラムです。BASICプログラム(BASICのSAVE命令で登録したプログラム)を呼び出そうとするとエラー(FILE MODE ERROR)になります。このときは CLS キーでエラーを解除してください。

③ファイルの削除

ファイルエリアに登録されているプログラム(ファイル)を削除します。

プログラムファイルのメニュー画面で K キーを押すと、ファイル名の入力待ちになります。

K

```
     << PROGRAM FILE >>

 Save  Load ➡Kill  Files
FILE NAME=?
```

削除したいプログラムのファイル名を入力して ↵ キーを押せば、削除が行われます。
〈例〉登録されているファイル名が"TEST"のプログラムを削除します。

TEST

```
     << PROGRAM FILE >>

 Save  Load ➡Kill  Files
FILE NAME=TEST_
```

↵

```
     << PROGRAM FILE >>

 Save  Load  Kill  Files
```

削除が行われ、メニュー画面に戻ります。

- ファイル名に、拡張子の指定を省略した場合は、".TXT"の指定とみなします。
- 指定したファイル名が見つからない場合はエラー(FILE NOT FOUND)になります。このときは CLS キーでエラーを解除してください。

④ファイル名の確認(Files)

プログラムファイルのメニュー画面で F キーを押すと、登録されているファイル名を表示し、最初のファイル名の左側に ➡ マークが表示されます。(何も登録されていないときは、画面は変わりません。)

F

(画面は、すでにこれらのプログラムが登録されている場合の例です。)

```
➡ABC           .TXT
 PRO           .TXT
 サンプルØØ1.BAS
```

表示されているファイル名以外にファイルがある場合は、↓、↑ キーで ➡ マークを移動させていくことにより画面に呼び出すことができます。

- このファイル名の確認画面で、SHIFT + M(LOAD) (または 2ndF M(LOAD))と操作すると、➡マークが示しているファイル名のプログラムを呼び出すことができます。

(8)BASICコンバータ(Basic)

BASICプログラム(中間コード形式)をTEXTプログラム(アスキー形式)に変換したり、逆にTEXTプログラムをBASICプログラムに変換したりする機能です。本機用のBASICプログラムをパソコン等でファイル、管理する場合などに利用します。

メインメニュー画面で B キーを押すと、BASICコンバータのメニュー画面になります。

B

```
<< BASIC CONVERTER >>

Basic←text   Text←basic
```

この画面で、BASICへの変換(Basic←text)、TEXTへの変換(Text←basic)を選ぶことができます。それぞれの頭文字に相当するキーを押してください。

①TEXTとBASICのプログラム変換(アスキー形式と中間コード形式の変換)

上記のメニュー画面で、B キーを押せばTEXTプログラムをBASICプログラムに変換し、プログラム・データエリアに入れます。

T キーを押せばBASICプログラムをTEXTプログラムに変換し、TEXTエリアに入れます。

〈例〉TEXTプログラムをBASICプログラムに変換します。

B

変換を実行します。その後、メインメニュー画面に戻ります。

```
<< BASIC CONVERTER >>

--- CONVERTING ---
```

(変換する内容が少ない場合、この実行中の画面は一瞬で終わります。)

●BASICプログラムに変換するとき、BASICのプログラム・データエリアにBASICプログラムがあった場合、あるいはTEXTプログラムに変換するときテキストエリアにTEXTプログラムがあった場合は、そのプログラムを削除して良いか聞いてきます。

[B]

```
 << BASIC CONVERTER >>

➡Basic←text   Text←basic
BASIC DELETE OK?  (Y)
```

[Y] キーを押せば、プログラムが削除され、変換が開始されます。

[Y] 以外のキーを押せば変換は中止され、メインメニュー画面に戻ります。

●通常、変換した場合でも変換元のプログラムは保持していますが、メモリ容量がたりない場合は、次のように、変換元のプログラムを削除して良いか聞いてきます。

```
 << BASIC CONVERTER >>

  --- CONVERTING ---
TEXT DELETE OK?  (Y)
```

[Y] キーを押せば、変換を行いながら変換元のプログラムを消去していきます。変換が終われば、変換元のプログラムはすべて消去されます。

[Y] 以外のキーを押せば変換は中止され、メインメニュー画面に戻ります。

ご注意

●パスワードが設定されているときは、メインメニュー画面から、BASICコンバータに入ることができません。あらかじめ、RUNまたはPROモードでパスワードを解除してください。

●BASICコンバータでは、データエリア(変数)を消去しません。フリーエリア(空きエリア)の容量が少ないと、変換できなくなることがありますので、あらかじめCLEAR命令で消去するなど、フリーエリアを広げておいてください。

●変換元のプログラムを消去しながら変換をしていっても、なおメモリがたりなくなるとエラー(MEMORY OVER)になります。
この場合、変換された部分と、変換できなかった部分がTEXTとBASICに分かれてしまいます。したがって、変換する前にプログラムをカセットテープなどに記録しておくことをお勧めします。
特にBASICからTEXTに変換するときに、起こる可能性が高いので注意してください。

●TEXTからBASICに変換するときは、TEXTの内容が何であっても、それを本機用のBASICプログラムと見なして変換作業を行います。したがってTEXTの内容によってはBASICに変換して、再度TEXTに変換しても、元のTEXT内容に戻らない場合もあります。

```
〈例〉 TEXT     1ØFORMULA
         ↓
       BASIC    1Ø:FOR␣MULA
         ↓
       TEXT     1ØFOR␣MULA
```

第6章　CASL

1．CASL(アセンブラ言語)

コンピュータ言語においてBASIC、FORTRAN、COBOLといった高水準な言語は、その記述がわかりやすいためプログラミングが容易であり、開発期間も短くて済みます。しかしこれらは、メモリの作業領域を多く必要としたり処理速度が遅いといった面があります。コンピュータの性能を最も効率よく活用できるのは機械語ですが、これは機械語コードで入力する必要があり、プログラミングは非常に手間がかかります。

アセンブラはこの機械語に最も近い言語で、BASICほど手軽ではないにしても、機械語に比べて使いやすく、コンピュータの性能を限界近くまで活用することができます。ただ、アセンブラ言語で効率の良いプログラムを作るには、各命令を実行するときに、レジスタやフラグなどが、どのような動きや働きをするのかを知る必要があります。これらの動き、働きなどはコンピュータのハードウェアに大きく左右され、メーカーや機種により異なるのが普通です。

このため通産省(通商産業省)の情報処理技術者試験では、COMETという仮想の計算機を設定し、これのアセンブラ言語としてCASLの仕様を取り決めた上で試験問題を出します。

本機のCASLモードは、このCASLの仕様に準拠したアセンブラ言語が扱えるため、情報処理技術者試験(第1種では必須、第2種では選択)のトレーニングとしてプログラミングからデバッグそしてシミュレーションまでの学習が容易に行えます。また、小型軽量であるため、ご家庭でも学校、職場、あるいは電車の中でも手軽にお使いいただけます。

また、CASLは従来の情報処理技術者試験で出題されていたアセンブラ言語CAP−Xと比較しますと、アドレスの与えかたが豊富になり、またスタック、入出力といった機能が付加され、最近の計算機の実状に合った仕様になっており、アセンブラ言語の入門用としても適しています。

本書は、本計算機の操作方法に重点をおいて説明していますので、COMETやCASLの仕様、CASLの文法、プログラミングについてのくわしい説明は、有名書店で数多く販売されている、情報処理技術者試験のCASLに関する書籍をご参照ください。

1．1　COMETの仕様概略

アセンブラ言語の理解を深めるため、本機のCASLモードでの仮想ハードウェア仕様(COMETに準拠)について概略を記しておきます。

(1)1語長　：16ビット

(2)制御方式　：逐次制御方式(ノイマン型)

(3)数値の表現：16ビットの2進数、負数は2の補数表示

(4)レジスタ　：①GRØ～4(16ビット)　General Register 汎用レジスタ

GR1～GR4は指標レジスタとしても使用。さらにGR4はスタックポインタとしても使用される。

②PC(16ビット)　Program Counter プログラムカウンタ

実行中の命令語の先頭アドレスを保持するレジスタ。

③FR(2ビット)　Flag Register フラグレジスタ

演算、比較結果の情報保持レジスタ。

	GRに設定されたデータ		
	負	ゼロ	正
FRの値	1Ø	Ø1	ØØ

1.2 命令語の構成

CASLの命令は2語長と定義されています。その構成をまとめます。

第1語				第2語	命令語とアセンブラとの対応		
OP					書きかた		
主	副	GR	XR	adr	命令コード	オペランド	ニモニックスペル
Ø	Ø				未使用		
1	Ø				LD	GR,adr,XR	load
	1				ST	GR,adr,XR	store
	2				LEA	GR,adr,XR	load effective address
2	Ø				ADD	GR,adr,XR	add arithmetic
	1				SUB	GR,adr,XR	subtract arithmetic
3	Ø				AND	GR,adr,XR	and
	1				OR	GR,adr,XR	or
	2				EOR	GR,adr,XR	exclusive or
4	Ø				CPA	GR,adr,XR	compare arithmetic
	1				CPL	GR,adr,XR	compare logical
5	Ø				SLA	GR,adr,XR	shift left arithmetic
	1				SRA	GR,adr,XR	shift right arithmetic
	2				SLL	GR,adr,XR	shift left logical
	3				SRL	GR,adr,XR	shift right logical
6	Ø	--			JPZ	adr,XR	jump on plus or zero
	1	--			JMI	adr,XR	jump on minus
	2	--			JNZ	adr,XR	jump on non zero
	3	--			JZE	adr,XR	jump on zero
	4	--			JMP	adr,XR	unconditional jump
7	Ø	--			PUSH	adr,XR	push effective address
	1			---	POP	GR	pop up
8	Ø	--			CALL	adr,XR	call subroutine
	1	--		---	RET		return from subroutine
9 ～ F					未使用		

1．3　命令の種類と機能

本機はCASLで定義された23の命令を持っています。以下に各命令の機能を説明します。なお、説明には次の表記法を使います。

①GR　　　　：GRの値を番号とする汎用レジスタを示します。（ただし、Ø≦GR≦4の範囲）

②XR　　　　：XRの値を番号とする指標レジスタを示します。（ただし、1≦XR≦4の範囲）

③SP　　　　：スタックポインタを示します。（スタックポインタはGR4を使用）

④adr　　　　：ラベル名に対応する番地、または10進の定数（ただし、−32768≦adr≦65535）を示します。adrはアドレスとして0～65535の値を持ちますが、32768～65535の値を負の10進定数で記述することもできます。（例：65535番地は−1と記述できます。）

⑤実効アドレス：adrとXRの内容とのアドレス加算値、またはその値が示す番地です。

⑥（X）　　　：X番地の内容、またはXが汎用レジスタ（GR）を示す場合は、そのGRの内容を表します。

⑦［　］　　　：［　］に囲まれた部分は省略可能です。XRを省略した場合は、指標レジスタによる修飾は行われません。

【各命令と機能】

⑴LD　　　GR，adr［，XR］

機能：（実効アドレス）をGRに設定します。

⑵ST　　　GR，adr［，XR］

機能：（GR）を実効アドレスが示す番地に格納します。

⑶LEA　　GR，adr［，XR］

機能：実効アドレスをGRに設定します。このときGRの値によりFRが設定されます。

例　：LEA　GR1，1Ø　　　　GR1に10を設定します。

LEA　GR2，Ø，GR3　　GR3をGR2に移動します。

LEA　GR3，−1，GR1　GR1から1を減じた値をGR3に設定します。

⑷ADD　　GR，adr［，XR］

機能：算術加算命令で、（GR）と（実効アドレス）の加算結果をGRに設定します。このときの演算結果によりFRが設定されます。

例　：ADD　GR2，WRK　　　（WRK番地）を（GR2）に加算します。

ADD　GR1，A1，GR3　A1とGR3とのアドレス加算結果の内容をGR1に加算します。

⑸SUB　　GR，adr［，XR］

機能：ADDの逆で、（GR）と（実効アドレス）の減算結果をGRに設定します。演算結果によりFRが設定されます。

⑹AND　　GR，adr［，XR］

⑺OR　　　GR，adr［，XR］

⑻EOR　　GR，adr［，XR］

機能：1語16ビットのビットごとの論理積（AND）、論理和（OR）、排他的論理和（EOR）を（GR）と（実効アドレス）で行い、GRに設定します。演算結果によりFRが設定されます。

例　：AND　GR2，MASK7　　MASK7を7にしておくと、GR2の下位3ビットが取り出せます。

EOR　GR1，FF　　FFが#FFFF(−1)の場合、GR1の全ビットが反転します。

arithmetic

(9)CPA　GR，adr [，XR]

(10)CPL　GR，adr [，XR]

logical

機能：(GR)と(実効アドレス)の算術比較(CPA)または論理比較(CPL)を行い、比較結果によりFRに次の値を設定します。

比較結果	FRの設定ビット値
(GR)>(実効アドレス)	ØØ
(GR)=(実効アドレス)	Ø1
(GR)<(実効アドレス)	1Ø

(11)SLA　GR，adr [，XR]

機能：(GR)を符号を除いて実効アドレスで指定したビット数だけ左シフトします。シフトの結果、空いたビット位置にはØが入り、FRを設定します。

(12)SRA　GR，adr [，XR]

機能：(GR)を符号を除いて実効アドレスで指定したビット数だけ右シフトします。シフトの結果、空いたビット位置には符号と同じものが入り、FRを設定します。

(13)SLL　GR，adr [，XR]

(14)SRL　GR，adr [，XR]

機能：(GR)を符号を含んで実効アドレスで指定したビット数だけ左シフト(SLL)、または右シフト(SRL)します。シフトの結果、空いたビットにはØが入り、FRを設定します。

(15)JPZ　adr [，XR]

(16)JMI　adr [，XR]

(17)JNZ　adr [，XR]

(18)JZE　adr [，XR]

機能：FRの値により実効アドレスに分岐します。分岐しないときは次の命令に進みます。

命　令	分岐するときのFRの値	意　味
JPZ	ØØ，Ø1	正またはØのとき分岐
JMI	1Ø	負のとき分岐
JNZ	ØØ，1Ø	Øでないとき分岐
JZE	Ø1	Øのとき分岐

(19)JMP　adr [，XR]

機能：FRの値に関係なく無条件に実効アドレスに分岐します。

(20)PUSH　adr [，XR]

機能：SPから1をアドレス減算してから実効アドレスを(SP)番地に格納します。

(21)POP　adr [，XR]

機能：(SP)番地の内容をGRに設定してからSPに1をアドレス加算します。

(22)CALL　adr［，XR］

機能：サブルーチンへジャンプする命令です。サブルーチンでの実行完了後の戻り番地確保のためにスタックを用います。

動作としてはSPから1をアドレス減算してから、PCの現在値に2をアドレス加算した値を(SP)番地に格納し、実効アドレスに分岐します。

(23)RET

機能：CALLに対する戻り命令です。(SP)番地の内容を取り出してからSPに1をアドレス加算し、先に取り出した番地に分岐します。

1．4　アセンブラの文法

本機のアセンブラ言語はCASLと同様に、疑似命令、マクロ命令、機械語命令を持っています。疑似命令、マクロ命令、機械語命令は次のように記述します。

ラベル欄	命令コード	オペランド欄	注釈欄
[ラベル]	START	[実行開始番地]	[注釈]
空　白	END	空　白	[注釈]
[ラベル]	DC	定　数	[注釈]
[ラベル]	DS	領域の語数	[注釈]
[ラベル]	IN	入力領域、入力文字長	[注釈]
[ラベル]	OUT	出力領域、出力文字長	[注釈]
[ラベル]	EXIT	空　白	[注釈]
[ラベル]	機械語命令	[命令の種類と機能]参照	[注釈]

表中、「空白」は記入してはならない箇所で、[ ]で囲まれた部分は省略可能を意味します。

1．5　疑似命令

疑似命令はアセンブラの制御、定数の定義、プログラムの連結のために必要なデータの生成などを行うもので、次の4種類があります。

(1)START：プログラムの先頭、実行開始番地の定義や、他のプログラムとの連結のための入り口名の定義をします。プログラムの最初には必要です。

(2)END　：プログラムの終わりの定義をします。プログラムの最後に必要です。

(3)DC　：define constantの略で、定数で指定したデータを格納します。

定数には次の4種があります。

①10進定数　　DC　n

nで指定した10進数値を1語の2進数データとして格納します。nは−32768～65535の範囲です。

②16進定数　　DC　＃h

hは4桁の16進数(Ø～F)です。hで指定された16進数値を1語の2進数データとして格納します。＃hは＃ØØØØ～＃FFFFの範囲です。

③文字定数　　DC '文字列'

文字列の左端から1文字ずつ連続する語の下位8ビット（第8～15ビット）に文字データを格納します。各語の第Ø～7ビットにはØのビットが入ります。文字列には266ページのキャラクタ・コード表の内、コード32～38および40～95の文字が使えます。なお、文字列の長さはØであってはなりません。

④アドレス定数　　DC　ラベル名

ラベル名に対応するアドレス値を1語の2進数データとして格納します。

(4)DS　　：define storageの略で、指定した語数の領域を確保します。
領域の語数は、10進定数(≧Ø)で指定します。Øの場合、領域は確保されず、ラベル名のみが有効となります。

1．6　マクロ命令

(1)IN　　：あらかじめ割り当てた入力装置から入力領域に、1レコードの文字データを入力します。本機ではカード読み取り装置からのデータを模擬的にキーから入力できます。なお、EOF(end of file)の入力は[↓]キーを用います。

(2)OUT　：出力領域に格納されている文字データを1レコードとして、あらかじめ割り当てた出力装置に出力します。本機では表示またはプリンタに出力できます。

(3)EXIT　：プログラムの実行を終了します。

(4)WRITE：この命令を実行するとレジスタGRØ～GR4およびFR、PCの内容を表示します。
また、プリンタが印字できる状態になっているときは、GRØ～GR3の内容を印字します。

本機のマクロ命令では、次の命令群を生成します。

		〈オブジェクト〉		〈命　令〉	
IN	a，b	7ØØØ	aaaa	PUSH	a
		7ØØØ	bbbb	PUSH	b
		8ØØØ	ØØØØ	CALL	Ø
		1244	ØØØ2	LEA	4，2，4
OUT	a，b	7ØØØ	aaaa	PUSH	a
		7ØØØ	bbbb	PUSH	b
		8ØØØ	ØØØ2	CALL	2
		1244	ØØØ2	LEA	4，2，4
EXIT		64ØØ	ØØØ4	JMP	4
WRITE		8ØØØ	ØØØ6	CALL	6

注）aaaa、bbbbはラベルaまたはラベルbが定義されているアドレスを示します。

これはØØØØH～ØFFFHに疑似OSとランタイムパッケージが存在し、ØØØØH、ØØØ2H、ØØØ4H、ØØØ6HがそれぞれIN、OUT、EXIT、WRITE処理ルーチンへのジャンプテーブルと仮定しています。参考にCASLモードのときの仮想のメモリマップを示します。

アドレス(16進)

ØØØØH

ジャンプテーブル	ØØØØ：JMP　XXX　ＩＮ処理アドレス ØØØ2：JMP　XXX　ＯＵＴ処理アドレス ØØØ4：JMP　XXX　ＥＸＩＴ処理アドレス ØØØ6：JMP　XXX　ＷＲＩＴＥ処理アドレス
疑似ＯＳ、ランタイムパッケージエリア	

ØFFFH

1ØØØH

ユーザープログラムのオブジェクトエリア

スタックエリア

XXXXH　← スタックポインタ

１．７　特殊命令

(1) ＊

機能：情報処理技術者試験の空欄穴埋め問題に対応するための、空欄入力命令です。

この命令をアセンブルしますと＃ØØØØが２語生成されます。シミュレーションのときには命令語の第１語が＃ØØＸＸ（ＸＸは何でもかまいません）の命令を実行後一時停止し、＊ＳＴＰ＊　と表示します。このときオブジェクトの変更やレジスタ値の確認をすることができます。その後、プログラムを続行することができます。

2. CASLモードの構成

CASLモードは次の3つの機能から構成されています。(168ページの機能一覧表を参照ください。)

アセンブラ　：TEXTエディタで作成したソースプログラムをオブジェクトコードに変換してメモリに書き込みます。
別売のプリンタCE-126Pが接続され、PRINTシンボルが点灯しているときは、アセンブルした結果を印字することができます。

モニタ　：レジスタの内容やオブジェクトコードを表示させたり、書き換えることができます。
ブレークポイントのアドレスを設定することができます。
また、DS命令等で確保したメモリの内容を書き換えることができます。

シミュレータ：オブジェクトプログラムを実行します。
ノーマル実行とトレース実行ができ、CE-126Pを接続すれば実行結果を印字することができます。
モニタ機能で指定したブレークポイントでの実行中断を行うことができます。

2.1　プログラム作成手順

CASLのプログラミングは次の手順で行われます。

①TEXTエディタ機能を使用してソースプログラムを入力します。
必要であればソースプログラムの印字、カセットテープへの記録を行います。

②CASLモードのアセンブラ機能によりオブジェクトコード(オブジェクトプログラム)を生成します。

③モニタ機能を使用して、オブジェクトプログラムに必要なブレークポイントのアドレス設定やレジスタ、プログラムカウンタなどの設定を行います。

④シミュレータ機能でオブジェクトプログラムのノーマル実行、またはトレース実行を行います。

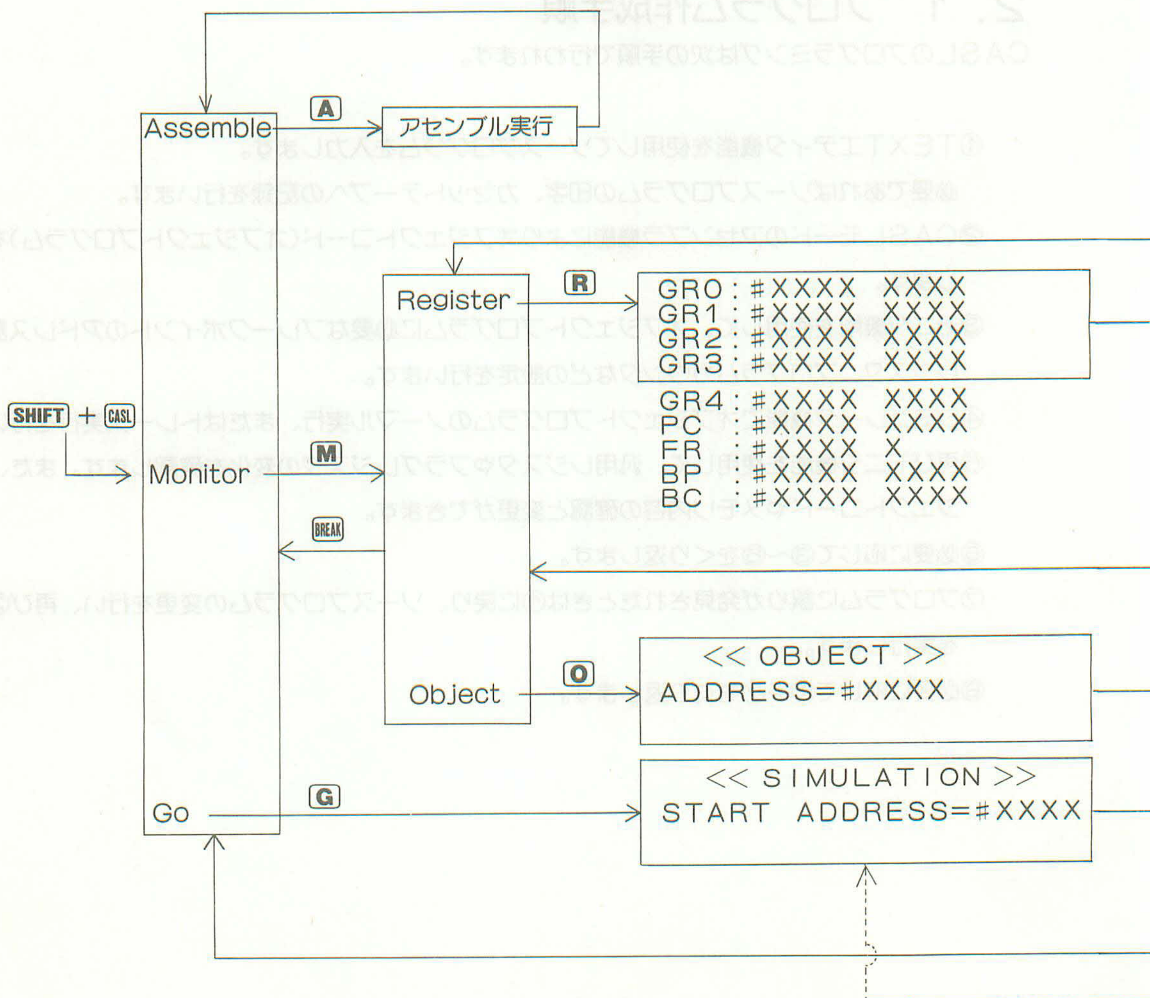
⑤再びモニタ機能を使用して、汎用レジスタやフラグレジスタの変化を確認します。また、オブジェクトコードやメモリ内容の確認と変更ができます。

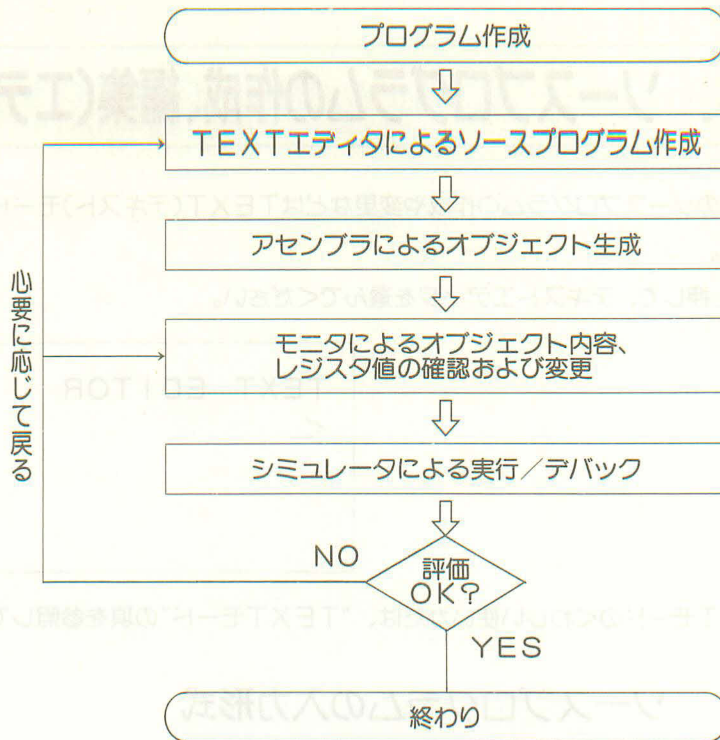
⑥必要に応じて③～⑤をくり返します。

⑦プログラムに誤りが発見されたときは①に戻り、ソースプログラムの変更を行い、再び②以降を実行します。

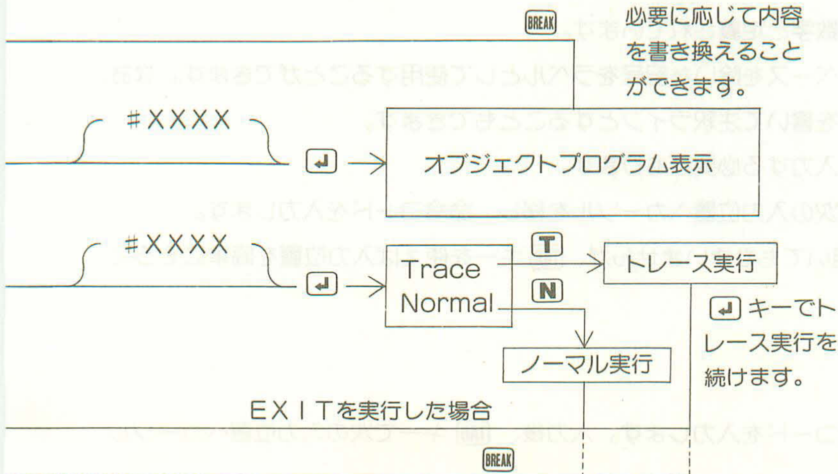
⑧必要に応じて①～⑦をくり返します。

CASLモード機能一覧





この表のいずれかの状態(プログラム実行中を除く)で TEXT キーを押すことによって、テキストエディタ(TEXTモード)に移ることができます。
なお、テキストエディタに移ったときはオブジェクトプログラムは消去されます。



3．ソースプログラムの作成、編集(エディット)

ＣＡＳＬのソースプログラムの作成や変更などはＴＥＸＴ(テキスト)モードのエディタ機能を用いて行います。

[TEXT] [E] と押して、テキストエディタを選んでください。

```
TEXT EDITOR
<
```

●ＴＥＸＴモードのくわしい使いかたは、“ＴＥＸＴモード”の項を参照してください。

3．1　ソースプログラムの入力形式

ソースプログラムの構成を以下に示します。

```
32767BGN  ADD    Ø, DAT, 1 ; SAMPLE
```

ライン№欄	ラベル欄(省略可)	命令コード欄	オペランド欄	注釈欄(省略可)
32767	BGN	ADD	Ø, DAT, 1	; SAMPLE

(1)ライン№欄

①ライン№は１～65279までの数値を使用できます。

②１～65279の範囲外の値を指定すると“ＬＩＮＥ　ＮＯ．　ＥＲＲＯＲ”と表示されます。

(2)ラベル欄

①ラベルは１～６文字まで使用でき、７文字以上を入力した場合は、先頭から６文字がラベルとして有効になります。

②ＣＡＳＬでは、ラベルは６文字以内で先頭の文字はアルファベット大文字、２文字目以降はアルファベット大文字または数字と定義されています。
本機では、コンマ(，)とスペースを除いた記号をラベルとして使用することができます。なお、ラベル欄にセミコロン(；)を書いて注釈ラインとすることもできます。

③ラベルはライン№に続けて入力する必要があります。

④ラベル入力後、[TAB] キーで次の入力位置へカーソルを移し、命令コードを入力します。
[TAB] の代わりに [SPACE] を用いてもかまいませんが、[TAB] キーを使えば入力位置を簡単にそろえることができます。

(3)命令コード欄

①アルファベットキーで命令コードを入力します。入力後、[TAB] キーで次の入力位置へカーソルを移します。

(4)オペランド欄

①オペランド欄はGRフィールド、adrフィールド、XRフィールドで構成されます。

②各ブロック(フィールド)の区切りには、必ずコンマ(，)が必要です。

③XRフィールドは省略することができます。

(5)注釈欄

①プログラムに注釈を付加する場合は、オペランドの直後にセミコロン(；)を入力し、その後に注釈を入力します。

●一行の長さは、注釈欄を含めて最大254文字まで入力することができます。

3．2　ソースプログラムの消去

TEXTモードの機能選択画面で、 D キーを押せばデリート(Del)機能が選ばれ、次のようにテキスト内容を消去(削除)して良いかきいてきます。(ただし、テキスト内容がない場合、画面は変わりません。)

```
TEXT DELETE OK? (Y)
```

このとき、 Y キーを押せば、テキスト内容がすべて消去され、TEXTモードの機能選択画面に戻ります。 Y 以外のキー(そのとき有効に働くキー)を押せば消去をせずに機能選択画面に戻ります。

3．3　ソースプログラムの入力

TEXTモードの機能選択画面で E キーを押せばテキストエディタが選ばれます。

```
TEXT EDITOR
<
```

このとき、 ↓ または ↑ キーを押すと、ソースプログラムなどのテキスト内容がある場合は、画面に表示されます。何もないときは、画面は変わりません。

新たにソースプログラムを入力するときなどは、 BREAK ON キーを押して機能選択画面に戻し、3.2 の方法でテキスト内容を消去してください。

次にソースプログラムの入力手順を説明します。

①ラインNo.を入力します。

②ラベルがないときは、 TAB キーを押します。カーソルが命令コード欄に移ります。

ラベルを入力するときは、ラインNo.に続けて入力して TAB キーを押します。

なお、 TAB キーの代わりに SPACE キーでスペースを入力してもかまいません。

③命令コードを入力します。その後 TAB キーを押せばカーソルがオペランド欄へ移ります。

④オペランドは、コンマ(,)で区切って入力します。

⑤注釈は、オペランドの後にセミコロン(;)を入力して、その後に入力します。
ただし、DC命令の文字列中のセミコロンは注釈のための区切りとしては扱われません。(文字として扱われます。)

⑥1行の入力を完了したら ↵ キーを押して、プログラムをメモリに格納します。↵ キーを押せばカーソルが消えます。

次の行を入力するときは、①からくり返します。

- 空欄穴埋め問題は＊命令を ＊ キーで入力します。
- すべての行の入力が終わったら BREAK ON キーで機能選択画面に戻します。

〈例〉ソースプログラムの入力

TEXT E	TEXT EDITOR
	<
1ØL1 TAB	1ØL1 _
START TAB	1ØL1 START _
L2	1ØL1 START L2_
↵	1ØL1 START L2
2ØL2 TAB	2ØL2 _
LD TAB	2ØL2 LD _
GR1, A	2ØL2 LD GR1, A_
↵	2ØL2 LD GR1, A

以下、同様の操作で引き続き右のプログラムを入力してください。以降の説明で本プログラムを活用します。

全ラインの入力完了後、BREAK ON キーで機能選択画面に戻ります。

なお、GRØ～GR4はGRを省略してØ～4のみを入力することができます。

```
3Ø      ADD    GR1, B
4Ø      ST     GR1, C
5Ø      EXIT
6ØA     DC     1Ø
7ØB     DC     2Ø
8ØC     DS     1
9Ø      END
```

4. アセンブル

ＴＥＸＴモードで入力したソースプログラムをオブジェクトコードに変換する(アセンブルする)場合、まず、ＣＡＳＬモードにします。

SHIFT ＋ CASL

メニュー画面になります。

```
*** CASL ***

Assemble Monitor Go
```

そして A キーを押します。

アセンブルが実行されます。

```
*** CASL ***

Assemble Monitor Go
complete !
```

アセンブル実行中は画面の下の行に“assembling”と表示され、終われば“complete !”と表示されます。アセンブルするプログラムが短いときは、瞬時に“complete !”と表示されます。

４．１　アセンブルリストをプリンタで印字する方法

本機にプリンタCE-126Pが接続されて電源が入っているとき、メニュー画面で SHIFT ＋ P↔NP と押すと画面右下にＰＲＩＮＴシンボルが点灯します。再度、SHIFT ＋ P↔NP と押すとこのシンボルは消えます。

ＰＲＩＮＴシンボルが点灯しているときにアセンブルを実行すると、アセンブルリストが印字されます。

```
ADD : OBJECT   : LINE NO.
               : 10
1000:1010 1008: 20
1002:2010 1009: 30
1004:1110 100A: 40
1006:6400 0004: 50
1008:000A      : 60
1009:0014      : 70
100A:0000      : 80
               : 90
|←①→|←― ② ―→|←― ③ ―→|

LABEL :ADDRESS  ┬
L1     1000     │
L2     1000     │
A      1008     ④
B      1009     │
C      100A     ┴
```

①アドレス(16進)

②オブジェクト(16進)

③対応するソースプログラムのラインNo.

④ラベルテーブル

アセンブルリストにはソースプログラムは印字されません。また、アセンブルエラーが発生したときはエラーが発生した行の前までが印字されます。

4．2　エラーメッセージ

アセンブルの実行で、ソースプログラムにエラーが検出されると、下表のようなエラーメッセージが表示されます。

エラーは CLS キーで解除して、TEXTエディタでソースプログラムを修正してください。

エラーの種類	エラーメッセージ	原　　因
オペコードエラー	OP−CODE　ERROR（ラインNo.）	ラインNo.で示すプログラムの命令コードに誤りがある
	OP−CODE　ERROR（Ø）	ソースプログラムがない
オペランドエラー	OPERAND　ERROR（ラインNo.）	ラインNo.で示すプログラムのオペランドに誤りがある
ラベルエラー	LABEL　ERROR（ラインNo.）	ラインNo.で示すプログラムのラベルに誤りがある
メモリエラー	MEMORY　ERROR（Ø）	●オブジェクトコードを記憶するメモリが不足している ●スタック用メモリが不足している
その他のエラー	OTHER　ERROR	プログラムの先頭にSTART命令がない。または、終わりにEND命令がない

5. モニタ

モニタ機能は、仮想計算機COMETの各レジスタの内容確認や、オブジェクトコードを表示することができます。

また、レジスタの値の設定・変更や、オブジェクトコードの設定・変更、およびシミュレートのためのブレークポイントの設定などを行うことができます。

モニタ機能はメニュー画面で M キーを押すことにより、選ぶことができます。

```
*** CASL ***

Assemble Monitor Go
```

M

```
<< MONITOR >>

Register   Object
```

モニタ画面では、次の操作が可能です。

操　作	機　　　能
[R]	レジスタの内容の表示と、レジスタへの値の設定ができます。
[O]	オブジェクトコードと、オブジェクトコードを逆アセンブルした命令の表示ができます。また、この内容を設定したり、変更したりできます。

5．1　レジスタの内容の表示

モニタ画面で [R] キーを押すとレジスタの内容やブレークポインタの内容が表示されます。

```
      << MONITOR    >>

   Register   Object
```

[R]

```
GRØ  #ØØØØ  Ø
GR1 :#ØØØØ  Ø
GR2 :#ØØØØ  Ø
GR3 :#ØØØØ  Ø
```

レジスタ　16進数で内容を表示　10進数で内容を表示

以降、[↓] キーでカーソル(このときのカーソルはコロン(:)の消灯)を下に移動していけば、GR4以降を画面に呼び出すことができます。

⋮

[↓]

```
GR4 :#1BØB  6923
PC  :#1ØØØ  4Ø96
FR  :#ØØØØ  Ø
BP   #FFFF  65535
```

注）GR4の値はメモリの使用状態によって変わります。

[↓]

```
PC  :#1ØØØ  4Ø96
FR  :#ØØØØ  Ø
BP  :#FFFF  65535
BC   #ØØØØ  Ø
```

戻すときは [↑] キーを押します。なお、[↓] キーの代わりに [↵] キーを押しても同様に働きます。

各レジスタ名は以下のとおりです。

表　示	レジスタ名	機　　能
GRØ～GR4	汎用レジスタ	算術演算と論理演算に用います。GR1～GR4は指標レジスタとしても用います。また、GR4はスタックポインタとしても用います。
PC	プログラムカウンタ	次に実行すべき命令が格納されているアドレスを記憶します。
FR	フラグレジスタ	特定の命令を実行した結果が、正、ゼロ、負になると、それぞれØ、1、2にセットされます。
BP	ブレークポインタ	シミュレーションでの実行を制御するために使用します。
BC	ブレークカウンタ	

5．2　レジスタに値を設定する方法

GRØ～GR4、PC、FR、BP、BCの各レジスタに値を設定することができます。

値を設定しようとするレジスタにカーソル(コロン(：)の消灯)を移し、値を10進数、16進数、ラベル、文字などで入力して ↵ キーを押せば、レジスタに値が設定されます。

〈例〉10進数の入力：　　123 ↵　#ØØ7B　123　が入力されます。

　16進数の入力：#ØØ7B ↵　#ØØ7B　123　が入力されます。

　ラベルの入力：　"L1" ↵　ラベル"L1"が定義されているアドレスが入力されます。

　文字の入力　：　'A' ↵　Aの文字コード　#ØØ41　65　が入力されます。

設定終了後、BREAK キーでモニタ画面に戻ります。

レジスタに値を設定するときは、次の点に注意してください。

①設定する値は10進数、16進数、ラベル、文字で入力します。

②負数は先頭に ー キーでマイナスを入力します。

③設定できる値の範囲は－32768～65535までの整数が可能です。この範囲を超えた値を入力して、↵ キーを押すと、入力前の値に戻ります。

④FRの値はØ、1、2しかとれません。Ø、1、2以外の値が入力されると、他のビットは無視されます。ただし、画面には入力した値が表示されます。

⑤アセンブル実行後の各レジスタには、以下の値が設定されています。

GRØ～GR3：Ø
GR4　：オブジェクトエリアの最上位番地＋1
PC　：メインプログラムの実行開始アドレス
　　(STARTで指定されたラベルのアドレス)
FR　：Ø
BP　：FFFF(16進)　65535(10進)
　　(ブレークポインタが設定されていない状態)
BC　：Ø

注）GR4はスタックポインタとして最大限に使えるように、オブジェクトエリアの最上位番地＋1に指定されていますが、この値はモニタ機能で呼び出してマニュアル操作で変更、またはプログラム上で変更することができます。

5．3　オブジェクトコードの表示

アセンブルの実行によって作成されたオブジェクトプログラムを表示させることができます。モニタ画面で [O] キーを押すとオブジェクトプログラムが記憶されているときは、アドレス＃１ØØØが表示されます。

```
        << MONITOR >>

Register     Object
```

[O]

```
        << OBJECT >>
ADDRESS=#1ØØØ
```

（ADDRESS=#1ØØØ：メインプログラムの開始アドレス）

次に [↵] キーを押すと、オブジェクトエリアの＃１ØØØ番地に記憶された内容から表示します。
表示内容は、オブジェクトコードと逆アセンブルしたニモニックです。

[↵]

```
第1語目→ 1ØØØ 1Ø1Ø LD    1，#1ØØ8
第2語目→ 1ØØ1:1ØØ8
第1語目→ 1ØØ2:2Ø1Ø ADD   1，#1ØØ9
第2語目→ 1ØØ3:1ØØ9
```

（各欄：アドレス／オブジェクトコード／オペランド／GRフィールド／アドレスフィールド／XRフィールド。GRフィールド・アドレスフィールド・XRフィールド：ニモニック表示）

以降 [↓] キーでカーソル(コロン(：)の消灯)を移動させていけば、以降のアドレスの内容を見ることができます。また、[↑] キーでアドレスを戻すことができます。
なお、この場合 [↓] の代わりに [↵] キーを使用しても同じ働きになります。

オブジェクトプログラムが記憶されていないときに、モニタ画面で [O] キーを押した場合はエラーになり“OBJECT　ERROE”が表示されます。
このときは [CLS] キーを押してメニュー画面に戻し、アセンブルを行ってください。

注）オブジェクトプログラムは、CASLモードから他のモード(TEXT、RUN)へ切り替わったときに消去されます。
したがって、他のモードからCASLモードに切り替えたときは、再度アセンブルを行ってください。

5．4　特定のアドレスに記憶された内容の表示

メインプログラムの開始アドレスを表示している画面で、表示させたいアドレスを10進数、16進数、またはラベルで入力し、⏎キーを押すと入力したアドレスの内容が表示されます。

入力したアドレスが偶数のときは、偶数番地を命令語の第1語目、次の番地を第2語目とみなして、逆アセンブルします。入力したアドレスが奇数の場合は、奇数番地を命令語の第1語目、偶数番地を第2語目とみなしますので注意してください。

```
<< OBJECT >>
ADDRESS=#1ØØØ
```

#1ØØ6 ⏎

```
1ØØ6  64ØØ  JMP   #ØØØ4
1ØØ7:ØØØ4
1ØØ8:ØØØA  *
1ØØ9:ØØ14
```

BREAKキーを押せば、モニタ画面に戻ります。

注1）逆アセンブルした結果、ニモニックで表示する内容がない場合は ＊ マークを表示します。
なお、オブジェクトプログラムがないアドレスは、ØØØØ　＊表示になります。

注2）入力できるアドレスは＃1ØØØから、アセンブル終了時にGR4に入っているアドレス－1の範囲です。

5．5　オブジェクトコードの書き換え

カーソル(コロン(：)の消灯)を書き換えたいアドレスへ移し、10進数、16進数、ラベル、文字でオブジェクトコードまたは、それに相当する内容を入力することにより、オブジェクトコードを書き換えることができます。

また、アセンブラの命令語も入力することができます。

● 10進数、16進数、ラベル、文字での入力の形式は176ページの“レジスタに値を設定する方法” を参照してください。

〈例〉アドレス＃1ØØØのオブジェクトコードを＃121Øに書き換えます。

アドレス＃1ØØØにカーソルを移してください。

```
1ØØØ 1Ø1Ø LD    1,#1ØØ8
1ØØ1:1ØØ8
1ØØ2:2Ø1Ø ADD   1,#1ØØ9
1ØØ3:1ØØ9
```

＃121Ø

```
1ØØØ 1Ø1Ø #121Ø_
1ØØ1:1ØØ8
1ØØ2:2Ø1Ø ADD   1,#1ØØ9
1ØØ3:1ØØ9
```

オブジェクトコードが書き換えられ
ニモニックも変わります。

```
1ØØØ 121Ø LEA   1,#1ØØ8
1ØØ1:1ØØ8
1ØØ2:2Ø1Ø ADD   1,#1ØØ9
1ØØ3:1ØØ9
```

続いてアセンブラの命令語を入力して書き換えます(元に戻します)。

LD TAB 1，＃1ØØ8

```
1ØØØ 121Ø LD 1,#1ØØ8_
1ØØ1:1ØØ8
1ØØ2:2Ø1Ø ADD   1,#1ØØ9
1ØØ3:1ØØ9
```

```
1ØØØ 1Ø1Ø LD    1,#1ØØ8
1ØØ1:1ØØ8
1ØØ2:2Ø1Ø ADD   1,#1ØØ9
1ØØ3:1ØØ9
```

6. シミュレータ

オブジェクトプログラムを実行する機能です。

メニュー画面で G キーを押すと、オブジェクトプログラムの実行処理に移ります。

```
*** CASL ***

  Assemble  Monitor  Go
```

G

```
  << SIMULATION >>
START ADDRESS=#1ØØØ
```

実行開始アドレス

実行開始アドレスは、START命令で指定されたアドレスが表示されます。START命令で実行開始アドレスが指定されていないときは、自動的に#1ØØØが実行開始アドレスになります。

表示されている実行開始アドレスを変更する場合は、上記のアドレス表示状態で、変更したいアドレスを10進数、16進数、またはラベルで入力します。

次に ↵ キーを押せば、実行方法選択画面になります。

```
  << SIMULATION >>
START ADDRESS=#1ØØØ

  Normal    Trace
```

実行方法選択画面では、次の操作が可能です。

操　作	機　　　能
N	ノーマル実行
T	トレース実行

6．1　ノーマル実行

実行方法選択画面で N キーを押せば、設定された実行開始アドレスからプログラムの実行を開始します。

EXIT命令が実行されると、メニュー画面に戻ります。

ノーマル実行中、次の場合にプログラムの実行を停止して、停止したアドレスを表示します。

- ブレークポインタ(BP)が設定されていて、プログラムカウンタ(PC)の値がブレークポインタの値と一致した場合。この場合、ブレークポインタが設定されているアドレスの命令は実行されていません。
- [BREAK] キーが押された場合。この場合、プログラムカウンタに停止(ブレーク)したときのアドレスが入っているので、[↵] キーを押せばプログラムの継続実行ができます。また、[CLS] や [BREAK] キーでメニュー画面に戻してから、再び実行操作をしても継続実行できます。
- ＊ 命令を実行した場合

  ＊ 命令により停止し、レジスタ内容などが表示されているときに、[↵] キーを押すと、＊ 命令をスキップし、次の命令を実行します。

  ＊ 命令(第一語目が＃ØØXX、第二語目が＃XXXXの命令語)を実行した場合、プログラムカウンタには第一語目のアドレスが入っています。したがって [BREAK] キーでプログラムを中断した後、オブジェクトの修正、または実行アドレスの修正を行わずにプログラムを再実行させますと、再度同じアドレスでプログラムが停止します。このような場合、命令語、または実行アドレスを修正した後、プログラムを実行してください。

  画面右下にPRINTシンボルが点灯しているときは、OUT命令による結果がプリンタで印字されます。

6.2 トレース実行

実行方法選択画面で [T] キーを押せば、表示されている実行開始アドレスからプログラムのトレース実行が開始されます。トレース実行中は1命令実行ごとに、実行結果(GRØ〜GR4、PC、FRの内容)を表示して停止します。ここで表示されるPCの値は命令実行後のPCの値、つまり次に実行する番地です。

次に [↵] を押すと、次の命令を実行して停止します。

```
       << SIMULATION >>
START  ADDRESS=#1ØØØ

    Normal    Trace
```

[T]

```
1ØØØ:  GRØ:ØØØØ  GR4:1BØB
       GR1:ØØØA  PC :1ØØ2
       GR2:ØØØØ  FR :ØØØØ
       GR3:ØØØØ  <LD>
```

[↵]

```
1ØØ2:  GRØ:ØØØØ  GR4:1BØB
       GR1:ØØ1E  PC :1ØØ4
       GR2:ØØØØ  FR :ØØØØ
       GR3:ØØØØ  <ADD>
```

実行した命令

メニュー画面で SHIFT ＋ P↔NP キーを押し、PRINTシンボルが点灯しているときは、以下のフォーマットで実行結果が(16進数で)印字されます。

```
ADD :GR0  GR1  GR2  GR3
1000:0000 000A 0000 0000
1002:0000 001E 0000 0000
1004:0000 001E 0000 0000
1006:0000 001E 0000 0000
|-①-+-②-+-③-+-④-+-⑤-|
```

①アドレス
②GRØ
③GR1
④GR2
⑤GR3

6.3 シミュレーションでのエラー

シミュレーションでは、次のエラーが発生することがあります。

エラーは CLS キーで解除してください。

エラーメッセージ	原　因
OBJECT ERROR	オブジェクトプログラムがない
＊MEM＊ ＊ERR＊	●JMP命令などで使用可能な領域を超えたアドレスへのジャンプが行なわれた。 ●使用可能な領域を超えて、データのロードやセーブをしようとした。
＊OPR＊ ＊ERR＊	OUT命令で出力文字を97文字以上にした。

7. CASL実行例

次のプログラムは5個のデータの総和を求めるプログラムです。

データは、ライン№130〜170に定義されています。また、ライン№120で総和を格納する領域を確保し、ライン№90でその領域に総和を収納しています。

```
1ØEXAM   START
2ØBGN    LEA    GRØ, Ø           (GRØをクリアする)
3Ø       LEA    GR1, Ø           (GR1をクリアする)
4Ø       JMP    AGN1             (無条件にAGN1へジャンプする)
5ØAGN    ADD    GRØ, DAT, GR1    (指標レジスタ(GR1)を利用してデータを
                                 順番にGRØへ加算する)
6Ø       LEA    GR1, 1, GR1      (GR1の内容に1を加えて、結果をGR1に
                                 入れる)
7ØAGN1   CPA    GR1, N           (GR1の内容とN番地の内容とを大小比較
                                 する)
8Ø       JMI    AGN              (FRの値が"負"を示していればAGNへジ
                                 ャンプする)
9Ø       ST     GRØ, TTL         (GRØの内容をTTL番地へ格納する)
1ØØ      EXIT                    (プログラムの実行終了)
11ØN     DC     5
12ØTTL   DS     1                (結果を格納するためのエリアを確保する)
13ØDAT   DC     #ØØØC           ┐
14Ø      DC     #Ø7F3           │
15Ø      DC     #Ø231           ├サンプルデータ
16Ø      DC     #ØØØ9           │
17Ø      DC     #ØØØF           ┘
18Ø      END                     (プログラム終了)
```

このプログラムをTEXTモードで入力し、CASLモードでアセンブルしてください。

7.1 オブジェクトコードの内容確認

以下の手順に従って、モニタ機能によりオブジェクトコードの内容を確認します。

操　作	表　示
	*** CASL *** Assemble Monitor Go
M	<< MONITOR >> Register Object
O	<< OBJECT >> ADDRESS=#1ØØØ

操　作	表　示
↵	1000 1200 LEA 0, #0000 1001:0000 1002:1210 LEA 1, #0000 1003:0000
↓ ⋮	1004:6400 JMP #100A 1005:100A 1006:2001 ADD 0, #1014, 1 1007:1014 1008:1211 LEA 1, #0001, 1 1009:0001 100A:4010 CPA 1, #1012 100B:1012 100C:6100 JMI #1006 100D:1006 100E:1100 ST 0, #1013 100F:1013 1010:6400 JMP #0004 1011:0004 1012:0005 * 1013:0000 1014:000C * 1015:07F3 1016:0231 ***** 1017:0009 1018:000F * 1019:0000
BREAK ON	<< MONITOR >> Register　Object

7.2 ノーマル実行

次にソースプログラムのラインNo.8Ø　JMI　AGN　にブレークポイントを設定して、プログラムの実行結果を調べることにします。

前項の方法でオブジェクトコードを調べると、この命令はアドレス#1ØØCに格納されていることがわかります。

操　作	表　　　　示	説　　明
	*** CASL *** Assemble Monitor Go	
[M]	<< MONITOR >> Register　Object	モニタ機能を選びます。
[R]	GRØ　#ØØØØ　Ø GR1:#ØØØØ　Ø GR2:#ØØØØ　Ø GR3:#ØØØØ　Ø	レジスタの内容を呼び出し、カーソル(コロン(:)の消灯)をブレークポインタレジスタへ移します。
[↓] : [↓]	BP　#FFFF　65535	ブレークポインタは設定されていません。
#1ØØC	BP　#FFFF　#1ØØC_	ブレークポインタをアドレス#1ØØCに設定します。
[↵]	BP　#1ØØC　41Ø8	
[↓]	BC　#ØØØØ　Ø	ブレークカウンタを2に設定します。
2[↵]	BC　#ØØØ2　2	
[BREAK][BREAK]	*** CASL *** Assemble Monitor Go	メニュー画面に戻します。
[G][↵]	<< SIMULATION >> START ADDRESS=#1ØØØ Normal　Trace	実行方法選択画面にします。
[N]	1ØØC:　GRØ:ØØØC　GR4:1AA5 *　*　GR1:ØØØ1　PC　:1ØØC *STP*　GR2:ØØØØ　FR　:ØØØ2 *　*　GR3:ØØØØ　<JMI>	ノーマル実行をします。#1ØØC番地でブレークがかかり、レジスタの内容を表示します。GRØにデータ#ØØØCが加えられ、GR1（カウンタ)には1が入っています。
[↵]	1ØØC:　GRØ:Ø7FF　GR4:1AA5 *　*　GR1:ØØØ2　PC　:1ØØC *STP*　GR2:ØØØØ　FR　:ØØØ2 *　*　GR3:ØØØØ　<JMI>	実行が再開され、次のデータ(#Ø7F3)をGRØに加え、GR1は2になります。
[BREAK][BREAK]	*** CASL *** Assemble Monitor Go	

操　作	表　　　　　示	説　　　明
[M][R]	GRØ　#Ø7FF　2Ø47 GR1：#ØØØ2　2 GR2：#ØØØØ　Ø GR3：#ØØØØ　Ø	モニタでレジスタを呼び出します。
[↓] ： [↓]	BC　　#ØØØØ　Ø	カーソルをブレークカウンタへ移します。
4[↵]	BC　　#ØØØ4　4	4を設定します。
[BREAK][BREAK]	*** CASL *** Assemble Monitor Go	メニュー画面に戻します。
[G][↵]	<< SIMULATION >> START ADDRESS=#1ØØC Normal　　Trace	実行方法選択画面にします。 実行開始アドレスは先程、停止したアドレスになっています。
[N]	1ØØC：　GRØ：ØA48　GR4：1AA5 *　　*　GR1：ØØØ5　PC　：1ØØC *STP*　GR2：ØØØØ　FR　：ØØØ1 *　　*　GR3：ØØØØ　<JMI>	ノーマル実行をします。 先に停止した状態からの継続実行になります。 GRØには5個のデータの合計が入っています。
[BREAK][BREAK]		メニュー画面に戻します。

注1）プログラムカウンタ(PC)とブレークポインタ(BP)の値が一致したときに、ブレークカウンタ(BC)の値がØまたは1であればプログラムの実行を停止し、ブレークカウンタの値をØにします。ブレークカウンタの値がØまたは1以外のときは、ブレークカウンタから1を減じて、プログラムを続行します。

注2）スタートアドレスとブレークポインタ(BP)の値が等しいとき、ブレークカウンタ(BC)の値がØであればプログラムを続行し、次にプログラムカウンタとブレークカウンタの値が一致したときに停止します。

また、ブレークカウンタの値が1であれば、何も実行せずに停止し、ブレークカウンタの値をØにします。ブレークカウンタの値がØおよび1以外のときは、ただちに1を減じてプログラムを続行します。

7．3 ブレークポインタの解除

設定したブレークポインタの解除は、次の2通りの方法で行えます。

①アセンブルの再実行

②ブレークポインタの設定時と同じように、メニュー画面より

[M][R][↓]…と操作して、BPレジスタを表示させます。次に−1 [↵] とキーを押すことにより解除できます。

[BREAK] キーでメニュー画面に戻します。

7．4 トレース実行

同じプログラムをトレース実行してみましょう。

ブレークポインタは解除してください。なお、次の例はアセンブル実行直後の例を示しています。

操　作	表　示	説　明
	*** CASL *** Assemble Monitor Go	
[G] [↵]	<< SIMULATION >> START ADDRESS=#1ØØØ Normal　Trace	
[T]	1ØØØ：GRØ：ØØØØ　GR4：1AA5 GR1：ØØØØ　PC　：1ØØ2 GR2：ØØØØ　FR　：ØØØ1 GR3：ØØØØ　<LEA>	トレースモードで実行を開始します。 LEA　Ø，ØでGRØをリセットします。
[↵]	1ØØ2：GRØ：ØØØØ　GR4：1AA5 GR1：ØØØØ　PC　：1ØØ4 GR2：ØØØØ　FR　：ØØØ1 GR3：ØØØØ　<LEA>	LEA　1，ØでGR1をリセットします。
[↵]	1ØØ4：GRØ：ØØØØ　GR4：1AA5 GR1：ØØØØ　PC　：1ØØA GR2：ØØØØ　FR　：ØØØ1 GR3：ØØØØ　<JMP>	JMP　AGN1に無条件にジャンプします。PCの値を見れば1ØØA番地であることがわかります。
[↵]	1ØØA：GRØ：ØØØØ　GR4：1AA5 GR1：ØØØØ　PC　：1ØØC GR2：ØØØØ　FR　：ØØØ2 GR3：ØØØØ　<CPA>	CPA　1，NでGR1の内容とN番地の内容を比較します。
[↵]	1ØØC：GRØ：ØØØØ　GR4：1AA5 GR1：ØØØØ　PC　：1ØØ6 GR2：ØØØØ　FR　：ØØØ2 GR3：ØØØØ　<JMI>	JMI　AGNでFRが2のときはAGNアドレスへジャンプします。
[↵]	1ØØ6：GRØ：ØØØC　GR4：1AA5 GR1：ØØØØ　PC　：1ØØ8 GR2：ØØØØ　FR　：ØØØØ GR3：ØØØØ　<ADD>	ADD　Ø，DAT，1で指標レジスタ(GR1)を利用してデータをGRØに加算します。
[↵]	1ØØ8：GRØ：ØØØC　GR4：1AA5 GR1：ØØØ1　PC　：1ØØA GR2：ØØØØ　FR　：ØØØØ GR3：ØØØØ　<LEA>	GR1に1を加算します。

操　作	表　　　　　　示	説　　明
↵ ⋮ ↵	1ØØA:GRØ:ØØØC　GR4:1AA5 GR1:ØØØ1　PC　:1ØØC GR2:ØØØØ　FR　:ØØØ2 GR3:ØØØØ　<CPA> 1ØØA:GRØ:ØA48　GR4:1AA5 GR1:ØØØ5　PC　:1ØØC GR2:ØØØØ　FR　:ØØØ1 GR3:ØØØØ　<CPA>	以降、同様な動作がデータ数分くり返されます。
↵	1ØØC:GRØ:ØA48　GR4:1AA5 GR1:ØØØ5　PC　:1ØØE GR2:ØØØØ　FR　:ØØØ1 GR3:ØØØØ　<JMI>	CPA　1，NでFRが1となったため、ジャンプせずに次の命令を実行します。
↵	1ØØE:GRØ:ØA48　GR4:1AA5 GR1:ØØØ5　PC　:1Ø1Ø GR2:ØØØØ　FR　:ØØØ1 GR3:ØØØØ　<ST>	ST　Ø，TTLでGRØの値を番地TTLへ格納します。
↵	1Ø1Ø:GRØ:ØA48　GR4:1AA5 GR1:ØØØ5　PC　:ØØØ4 GR2:ØØØØ　FR　:ØØØ1 GR3:ØØØØ　<JMP>	EXIT命令により実行を停止します。(EXIT処理ルーチンへジャンプ)
↵	*** CASL *** Assemble Monitor Go	メニュー画面にもどります。

7.5 空欄穴埋め問題の実行例

前の例題でプログラムラインNo.60が空欄穴埋め問題となっていた場合、下の例のように60ラインに＊を入力します。

```
10EXAM  START
20BGN   LEA   GR0,0
30      LEA   GR1,0
40      JMP   AGN1
50AGN   ADD   GR0,DAT,GR
  1
60      LEA   GR1,1,GR1
70AGN1  CPA   GR1,N
80      JMI   AGN
90      ST    GR0,TTL
100     EXIT
110N    DC    5
120TTL  DS    1
130DAT  DC    #000C
140     DC    #07F3
150     DC    #0231
160     DC    #0009
170     DC    #000F
180     END
```

```
10EXAM  START
20BGN   LEA   GR0,0
30      LEA   GR1,0
40      JMP   AGN1
50AGN   ADD   GR0,DAT,GR
  1
60      *
70AGN1  CPA   GR1,N
80      JMI   AGN
90      ST    GR0,TTL
100     EXIT
110N    DC    5
120TTL  DS    1
130DAT  DC    #000C
140     DC    #07F3
150     DC    #0231
160     DC    #0009
170     DC    #000F
180     END
```

このプログラムをアセンブルするとラインNo.60に対応するオブジェクトコードが"＃ØØØØ　＃ØØØØ"になります。

```
ADD : OBJECT   : LINE NO.
               : 10
1000:1200 0000: 20
1002:1210 0000: 30
1004:6400 100A: 40
1006:2001 1014: 50
1008:0000 0000: 60
100A:4010 1012: 70
100C:6100 1006: 80
100E:1100 1013: 90
1010:6400 0004: 100
1012:0005      : 110
1013:0000      : 120
1014:000C      : 130
1015:07F3      : 140
1016:0231      : 150
1017:0009      : 160
1018:000F      : 170
               : 180
```

このプログラムを実行すると、＃1ØØ8番地でプログラムの実行を停止しますので、レジスタの値を確認することができます。また、モニタ機能を使って、オブジェクトの確認や修正をすることができます。

操　　作	表　　　示	説　明
	*** CASL *** Assemble Monitor Go	
G ↵	<< SIMULATION >> START ADDRESS=#1ØØØ Normal　Trace	
N	1ØØ8:　GRØ:ØØØC　GR4:1AAC *　　*　GR1:ØØØØ　PC　:1ØØ8 *STP*　GR2:ØØØØ　FR　:ØØØØ *　　*　GR3:ØØØØ　<*>	*命令があるとプログラムの実行を停止します。
↵	1ØØ8:　GRØ:ØØ18　GR4:1AAC *　　*　GR1:ØØØØ　PC　:1ØØ8 *STP*　GR2:ØØØØ　FR　:ØØØØ *　　*　GR3:ØØØØ　<*>	注1) 注2)
BREAK BREAK	*** CASL *** Assemble Monitor Go	
M O #1ØØ8	<< OBJECT >> ADDRESS=#1ØØ8_	アドレス#1ØØ8のオブジェクトプログラムを呼び出します。
↵	1ØØ8　ØØØØ　* 1ØØ9:ØØØØ 1ØØA:4Ø1Ø　CPA　1,#1Ø12 1ØØB:1Ø12	
LEA GR1, 1,GR1	1ØØ8　ØØØØ　LEA　GR1,1,GR1_ 1ØØ9:ØØØØ 1ØØA:4Ø1Ø　CPA　1,#1Ø12 1ØØB:1Ø12	空欄の答えを入力します。
↵	1ØØ8　1211　LEA　1,#ØØØ1,1 1ØØ9:ØØØ1 1ØØA:4Ø1Ø　CPA　1,#1Ø12 1ØØB:1Ø12	アセンブルし、オブジェクトが生成されます。
BREAK BREAK G "BGN"	<< SIMULATION >> START ADDRESS="BGN"_	もう一度初めからプログラムを実行します。
↵	<< SIMULATION >> START ADDRESS=#1ØØØ Normal　Trace	
N	*** CASL *** Assemble Monitor Go	終了するとメニュー画面に戻ります。
M O "TTL" ↵	1Ø13　ØA48　***** 1Ø14:ØØØC 1Ø15:Ø7F3　***** 1Ø16:Ø231	注3)

注1）シミュレーションで*命令を実行し、プログラムが停止しているときに ↵ キーを押すと、次の命令から実行を開始します。実行後、*命令があると再び停止します。

注2）この時点でGRØには#Ø7FF、GR1には#ØØØ1が入っているべきですが、値が異なっていることがわかります。

注3）合計を確認すると正しい値が入っています。つまり、空欄に入れた答えが正しいことがわかります。この時点では、オブジェクトコードは変更されていますが、ソースプログラムは変更されていません。

8. メモリ領域

ソースプログラム、オブジェクトプログラムはそれぞれ下図の領域に格納されます。

機械語エリア （本機のCPUによって動作する機械語）
ファイルエリア （プログラムファイル）
TEXTエリア （ソースプログラム）
BASICプログラムエリア
CASL （オブジェクトプログラム）
BASIC変数

この図の領域は本機のメモリ上に常に存在しているわけではありません。機械語の領域以外は、使用したときに自動的に確保されます。

8．1　格納できるステップ数の計算方法

ソースプログラム入力後、BASICのRUNモードにしてFRE ↵ と押すと、オブジェクトコードの格納可能な領域の大きさがバイトの単位の数値で表示されます。

オブジェクトコードは1ステップ16ビット（2バイト）であるため、この領域に格納できるステップ数は次式で計算できます。

$$\mathrm{INT}\left(\frac{\text{FRE}-\text{ラベル数}\times 8-4}{2}\right)$$

ただし、INTはカッコ内の計算結果の整数部を値とします。この計算式で求めた値にはスタックエリアも含まれています。スタックエリアが31ステップ以下になるとエラーになります。

〈例〉

```
L1  START  L2
L2  LD     GR1，A
    ADD    GR1，B
    ST     GR1，C
    EXIT
A   DC     1Ø
B   DC     2Ø
C   DS     1
    END
```

このプログラムの場合、オブジェクトサイズは7ステップとなります。

9. 仮想計算機COMETとの相異点

本機は通産省の設定している仮想計算機COMETと、そのアセンブラ言語の仕様に以下の相異点があります。

⑴START命令

ラベルが省略できます。また、オペランドが省略された場合、＃1ØØØ番地からプログラムを実行します。

⑵DC命令

オペランドが10進定数で−32768～65535の範囲にないときは、アセンブルするとエラーになります。

⑶IN命令(CALL　＃ØØØØ)

IN命令を実行すると、画面に“?”が表示され、キーからの入力が可能になります。このとき、[↓]キーの入力があると、入力文字長に−1(＃FFFF)を設定します。

この機能は、入力装置としてカードリーダを想定したときに利用すると便利です。

⑷OUT命令(CALL　＃ØØØ2)

OUT命令を実行すると、PRINTシンボルが点灯しているときは印字し、点灯していないときは画面に表示します。

出力文字数が97文字以上のときはエラーになります。

⑸WRITE命令(CALL　＃ØØØ6)

WRITE命令を実行するとCOMETのレジスタ内の内容を表示し、プログラムの実行を停止します。このとき[↵]キーを押せばプログラムの実行を続行します。

⑹複数プログラムの連結

本機には、別々にアセンブルして作成したオブジェクトプログラムを連結する“リンカ”の機能はありません。このような動作をさせたいときは、個々にSTART命令とEND命令で囲んだ複数のプログラムを一度にアセンブルしてください。なお、アセンブルするとき、他のプログラムのラベルと同じものがあるとエラーになります。

第7章　機械語モニタ

プログラム言語は用途により色々な種類があります。本機では、高級言語であるＢＡＳＩＣと機械語が使用できるようになっています。
ＢＡＳＩＣは英語に近い形で使いやすい言語ですが、処理スピードの点では機械語に劣ります。一方機械語は人間にとっては、非常にわかりにくい言語ですが、色々なことができ、コンピュータの機能を最大限に活用することができます。
本機はこの機械語を使用するために機械語モニタ機能が内蔵されています。
機械語モニタは、従来のポケットコンピュータではＢＡＳＩＣ命令のＰＥＥＫ、ＰＯＫＥ、ＣＡＬＬを使用しなければならなかった機械語の入出力や実行を簡単な命令だけで取り扱えるようにしたシステムプログラムです。
なお本機はＣＰＵ(シーピーユー：中央演算処理装置)に、優れた８ビットＣＰＵとして広くパソコン等で利用されているＺ８０(ＣＭＯＳ　Ｚ８０Ａ相当品)を使用しています。
Ｚ８０は人気のあるＣＰＵであり、入門書等の関連書籍が豊富に市販されていますので、Ｚ８０機械語についてはそれらの書籍をご参照ください。
本章では機械語モニタの各命令(コマンド)の働きについて説明します。

1．機械語モニタを使う上でのきまり

```
MACHINE LANGUAGE MONITOR
*
```

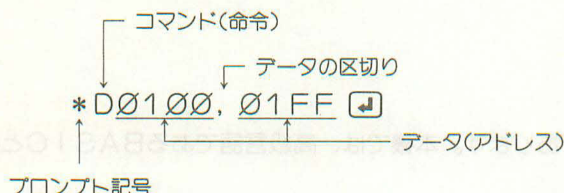
機械語モニタは機械語モニタモードで使います。BASICモード（RUNまたはPROモード）でMON ↵ と押せば、機械語モニタモードに切り替わり、右のように表示されます。

（パスワードが設定されているときは、機械語モニタモードにすることはできません。）

＊記号は機械語モニタモードのプロンプト記号です。

コマンド（命令）は必ずこのプロンプト記号の後に書き、アドレス（番地）やデータの指定が必要な場合は、コマンドに続けて指定します。そして ↵ キーを押し、実行させせます。

〈例〉



- ●アドレスおよびデータの指定は16進数で行います。
- ●アドレスおよびデータの区切りはコンマ（，）で指定します。
- ●アドレスおよびデータの指定にØ～Fの16進数字とコンマ以外を入れるとエラー（SYNTAX ERROR）になります。
- ●機械語モニタモードは、BASIC、TEXT キーの操作および電源を切ることで解除されます。

ご注意

①機械語は大変難解なためプログラム作成において誤りが発生しやすくなります。このため、実行させますと機械語プログラム、BASICプログラムおよびデータなど計算機が記憶している内容を壊してしまう恐れがあります。したがって、BASICプログラム等はカセットテープに記録しておいてください。また、機械語プログラムも実行する前に、CSAVE M命令（BASICの命令）でテープに記録しておくことをお勧めします。

②機械語モニタ機能で、機械語エリア（USERコマンドで確保したエリア）以外のアドレスを使用しますと、BASICプログラムやTEXTプログラムなど、他の記憶内容を壊したり、正しい動作をしないことがありますので、確保した機械語エリアのみを使用してください。

2. 機械語モニタの各命令の説明

＊USER……ユーザーエリア

機　能　機械語エリアの確保および確保したエリアの表示を行います。

書式例　(1)　USERØ1FF ↵

(2)　USER ↵

(3)　USERØØFF ↵

説　明
- 書式例(1)を実行するとメモリのØ1ØØH番地(開始番地)からØ1FFH番地(終了番地)までを機械語エリアとして確保します。開始番地は自動的にØ1ØØH番地になります。

```
＊USERØ1FF
FREE：Ø1ØØ−Ø1FF    ←確保した範囲
＊
```

- 書式例(2)を実行すると、現在確保されている機械語エリアの番地を表示します。

```
＊USER
FREE：Ø1ØØ−Ø1FF
＊
```

機械語エリアが確保されていないときは"FREE：NOT　RESERVED"と表示されます。

- 書式例(3)を実行すると、機械語エリアを消去します。そして、次の表示をします。
  USERØØFF ↵
  "FREE：NOT　RESERVED"
- 確保できない範囲(機械語エリアとして使用できない範囲)を指定するとエラー(MEMORY　ERROR)になります。

＊S……………セットメモリ

機　能　メモリの内容を書き換えます。

書式例　(1)　SØ1ØØ ↵

(2)　S ↵

説　明
- 書式例(1)を実行すると、Ø1ØØH番地(開始番地)の現在のメモリ内容が表示され、変更するデータの入力待ちになります。

```
＊SØ1ØØ
Ø1ØØ：1Ø−
       ↑
   現在のメモリ内容
```

- データを変更するときは、1バイトデータ(2桁の16進の数字)を入力し、↵ キーを押します。データが変更され、次のアドレス(番地)のデータ入力待ちになります。

データを変更しない場合は、データを入力せずに ↵ キーを押します。そのまま次のアドレスのデータ入力待ちになります。

- データは2桁以上入力できません。入力途中の値を取り消すときは ◄ キーまたは CLS キーを押します。
- ↓ キーで、次のアドレスを呼び出すことができます。また ↑ キーで前のアドレスを呼び出すことができます。
- 書式例(2)を実行すると、以前のSコマンドで表示していた次のアドレスのメモリ内容が呼び出されます。
- BREAK キーを押せばコマンド(命令)待ちの状態に戻ります。

〈例〉次のオブジェクトプログラムをØ1ØØH番地から書き込みます。

```
〈オブジェクトプログラム〉
3E  Ø1
18  Ø4
3A  ØF  Ø1
3C
32  ØF  Ø1
C9
```

＝参考＝　上記オブジェクトプログラムのソースプログラムを示します。

```
〈ソースプログラム〉
       LD    A, Ø1H          ←AレジスタにØ1Hを入れる
       JR    ST              ←ラベルSTの番地へジャンプ
       LD    A, (Ø1ØFH)      ←AレジスタにØ1ØFH番地の内容を入れる
       INC   A               ←Aレジスタの内容に1を加算する
ST:    LD    (Ø1ØFH), A      ←Aレジスタの内容をØ1ØFH番地へ入れる
       RET                   ←リターン
```

機械語プログラムは、通常アセンブラ言語でソースプログラムを作成し、それをハンドアセンブル(手操作によりオブジェクトコードへ置き換えること)します。そして、Sコマンドを利用して計算機に書き込みます。

注）本機には、Z80のソースプログラムをオブジェクトプログラムにアセンブルする機能はありません。

〈操　作〉	〈表　示〉	
BREAK		Ø1ØØH番地からØ1FFH番地までを機械語エリアとして確保（余裕のある大きさです）
USERØ1FF ↵	＊USERØ1FF	
	FREE：Ø1ØØ－Ø1FF	
SØ1ØØ ↵	＊SØ1ØØ	
3E ↵	Ø1ØØ：1Ø－3E	
Ø1 ↵	Ø1Ø1：12－Ø1	
18 ↵	Ø1Ø2：ØØ－18	
Ø4 ↵	Ø1Ø3：ØØ－Ø4	

3A ↵	Ø1Ø4:Ø1-3A
ØF ↵	Ø1Ø5:1Ø-ØF
Ø1 ↵	Ø1Ø6:5Ø-Ø1
3C ↵	Ø1Ø7:43-3C
32 ↵	Ø1Ø8:2D-32
ØF ↵	Ø1Ø9:47-ØF
Ø1 ↵	Ø1ØA:38-Ø1
C9 ↵	Ø1ØB:3Ø-C9
BREAK	Ø1ØC:31-
	*

ここには以前の内容が表示されるため、このとおりの数値になるとは限りません。

＊D……………ダンプメモリ

機　能　メモリの内容を表示させます。

書式例　(1)　DØ1ØØ ↵

(2)　D ↵

(3)　DØ1ØØ, Ø1FF ↵

説　明　●書式例(1)を実行すると、Ø1ØØH番地(開始番地)から16バイト分のメモリの内容（Ø1ØØH～Ø1ØFH番地の内容)が表示されます。(プリントモードでは、印字をします。)

〈表示例〉

16バイト単位の先頭アドレス→　チェックサム→

```
Ø1ØØ  :  3E  Ø1  18  Ø4   >...
(1D)     3A  ØF  Ø1  3C   :..<
         32  ØF  Ø1  C9   2..
         31  ØØ  ØØ  ØØ   1...
```

各データをアスキーコードとした場合の該当文字を表示します。ただし、ØØH～1FHのデータはピリオド(.)で表示されます。

●アドレス(番地)の区分は規定(×××ØH～×××FH番地に規定)されていて、その区分内であればどのアドレスを指定しても同じになります。たとえばØ1Ø4H番地を指定した場合、Ø1Ø4H番地を含む16バイト、つまりØ1ØØH～Ø1ØFH番地を指定したことになります。

●↓キーを押せば、次の16バイト分のメモリ内容が表示されます。↑キーを押せば、前の16バイト分のメモリ内容が表示されます。

●書式例(2)を実行すると、以前にDコマンドで表示していた続きを表示します。

●書式例(3)を実行すると、プリントモード時はØ1ØØH(開始番地)～Ø1FFH（終了番地)のメモリ内容を印字します。印字が終了すれば、コマンド待ちの状態に戻ります。プリントモードでないときは、Ø1ØØH番地(開始番地)から16バイト分のメモリの内容を表示します。(この場合、終了番地の指定は意味がなくなります。)

●プリントモードの指定／解除は、次項のPコマンドまたは SHIFT ＋ P↔NP キーで行います。

● BREAK キーを押せばコマンド(命令)待ちの状態に戻ります。

チェックサム…検査合計　データ項目の集まりの合計であって、そのデータが記録されるとき計算され、検査の目的でその集まりにつけられるもの。本機では、Dコマンドの実行によって表示された16バイト分のメモリの内容を合計し、その合計値の下位1バイトをチェックサムとして表示します。

プログラムをマニュアル操作で入力したときなど、このチェックサムが元のプログラムと一致することを確認することにより、表示された16バイトの中に入力誤りがないかどうかを確認できます。ただし、入力誤りが2個所以上あった場合は、偶然チェックサムが一致してしまうことがあります。

＊P……………プリントスイッチ

機　能　プリントモードの設定／解除を行います。

書式例　P ↵

説　明　●P ↵ と押すたびに、プリントモードの設定と解除が交互に行われます。(プリントモードでは画面右下に"PRINT"シンボルが点灯します。)

SHIFT ＋ P↔NP キーの操作と同じです。

●プリンタが接続されていないときや、プリンタの電源が入っていないときは、Pコマンドは働きません。

＊G……………ゴーサブ

機　能　指定したアドレス(番地)から機械語プログラムを実行します。

書式例　GØ1ØØ ↵

説　明　●BASICのGOSUB命令と同様で、指定したアドレスから機械語プログラムを実行し、機械語のリターン命令があればコマンド待ちの状態に戻ります。

●必ずプログラムの最後にRET(リターン)命令を入れてください。

RET命令がないと暴走します。

暴走…でたらめな実行、または一定の実行を続けて止まらなくなること。内部の状態によって、どのような動きをするか分からず、多くの場合、機械語プログラムやBASICプログラム、データなど記憶内容を破壊します。

●プログラムの実行を中止する場合は、リセット(RESET)スイッチ❺を押します。

注)機械語プログラムは1カ所でもまちがいがあると暴走をすることがあります。このため、BASICなど、他のプログラムが計算機に入っているときは、あらかじめカセットテープに記録しておいてください。

〈例〉Sコマンドの例で入力したプログラムを実行する場合

G0100 ↵

```
*G0100
*
```

すぐにコマンド待ちの状態に戻ります。

010FH番地に01Hが入っていますので、Dコマンドで確認します。

D0100 ↵

```
0100 :  3E 01 18 04   >...
 (1E)   3A 0F 01 3C   :..<
        32 0F 01 C9   2..
        31 00 00 01   1...
                 ↑
            01Hが入っています
```

次に0104Hから実行します。

BREAK G0104 ↵

```
*G0104
*
```

010FH番地に1が加えられています。Dコマンドで確認します。

D0100 ↵

```
0100 :  3E 01 18 04   >...
 (1F)   3A 0F 01 3C   :..<
        32 0F 01 C9   2..
        31 00 00 02   1...
                 ↑
            1が加えられています
```

くり返し0104H番地から実行すれば、その都度010FH番地に1が加えられます。

*R……………リードSIO

機　能　SIO(シリアル入出力装置)に送られてくるデータを読み込みます。パソコン等との機械語の通信等に使用します。

書式例　(1)　R ↵

(2)　R0100 ↵

説　明　SIOからインテル・ヘキサ形式でデータを読み込みます。

- 書式例(1)は読み込んだデータを受信情報の中で示される番地へ入れていきます。
- 書式例(2)は読み込んだデータを0100H番地(ロードアドレス)から順番に入れていきます。
- 読み込みが終了すると、データが入ったエリア(領域)を表示します。
- 読み込みを中止するときは、コマンド待ちの状態になるまで、BREAK キーを押し続けてください。

＊W……………ライトＳＩＯ

機　能　ＳＩＯ(シリアル入出力装置)にデータを出力します。パソコン等との機械語の通信等に使用します。

書式例　WØ1ØØ，Ø1FF ↵

説　明　●書式例を実行するとØ1ØØH番地(開始番地)からØ1FFH番地(終了番地)までのメモリ内容をＳＩＯからインテル・ヘキサ形式で出力します。

●出力を中止するときは、コマンド待ちの状態になるまで、BREAK キーを押し続けてください。

注）周辺機器接続端子(11ピン)❷にプリンタを動作状態で接続し、このWコマンドを実行しますと本機およびプリンタが誤動作をすることがあります。この場合、プリンタの電源を切り、本機の BREAK キーを押し続けてください。

3. 機械語モニタモードでのエラー表示

機械語モニタモードでは以下のエラー表示が行われます。
エラーが発生したときは CLS キーで解除してください。

エラーメッセージ	内　　　容
SYNTAX　ERROR	文法的に実行できない
MEMORY　ERROR	機械語エリアとして使用できる範囲を外れて、機械語エリアを確保しようとした
I／O　DEVICE　ERROR	ＳＩＯに対するリードインエラー、チェックサムエラーなど
OTHER　ERROR	その他のエラー

第 8 章
BASICの各命令の説明

この章ではBASICの各命令を1つ1つ説明します。

まず、本機がどのような働きをする命令を持っているかを見ていただき、その後プログラムを作成するうえで必要な命令の説明からお読みいただければよいと思います。

次に、以降の説明で書式などに使用する用語の意味を示します。

式……………数値、数値変数およびこれらを含んだ計算式を示します。

変数…………配列要素を含んだ数値変数、文字変数を示します。

"文字"……"　"(ダブルクォーテーションマーク)で囲まれたキャラクタ(文字、数字、記号)を示します。

文字列………"文字"、文字変数を示します。

(　)………カッコでくくる必要があることを示します。

[　]………省略可能であることを示します。ただし、この後にほかの指定を行う場合は、コンマなどの区切り記号が必要です。

{A / B}………AあるいはBを選択することができます。

マニュアル…マニュアル操作による実行が可能です。

プログラム…プログラムによる実行が可能です。

省略形………各命令は、そのつづりを省略した形で入力できるものがあり、その場合は一番省略した形で示しています。

〈例〉(省略形…P.)　これはPRINTの省略形ですが、次の形でも有効になります。

PR.
PRI.
PRIN.

注1) 変数などに続いてBASIC命令を入力する場合は、変数と命令の間にスペースを入力してください。

〈例〉 5Ø IF A=B THEN 1ØØ
　　　　　　　　↑スペースを入れる。

注2) 関数命令等をマニュアルで実行する場合はRUNモードで実行してください。

PROモードで実行するとエラー12になります。

ただし、PRINT命令の中で使用すれば、PROモードでも実行できます。

〈例〉 PRINT CHR$ 9Ø↵ →　Z

関数

1 AND…………アンド

プログラム
マニュアル

省略形……AN.

機能 式と式との論理積を計算します。また、条件式の結合を行います。

書式 式 AND 式

条件式 AND 条件式

参照 OR、NOT、IF

説明 ●2進数において、論理積は次のような値を取ります。

1 AND 1=1

1 AND Ø=Ø

Ø AND 1=Ø

Ø AND Ø=Ø

なお、10進数の論理積を求めた場合でも、計算機内では10進数を2進数に変換したうえで、各桁の論理積を求め、その結果を10進数に戻します。

たとえば、41と27の論理積は次のように計算されます。

41 AND 27=9

```
      101001………41
AND<
      011011………27
      ──────
      001001………9
```

41と27をそれぞれ2進数に変換し、各桁のANDを取ります。そして、その結果を10進数に変換すれば9になります。

●式の値は−32768〜32767の整数部が有効になります。

●2つ以上の条件をすべて満足するような条件を1つの式で表します。

〈例〉 A>Ø AND A<6　　AはØよりも大きく、かつ6よりも小さい。

5Ø IF B>5 AND C>=4 THEN………

もし、Bが5よりも大きく、かつCが4よりも大きいか等しいとき、THENに続く命令を実行します。

関数

2 ASC…………アスキー

プログラム
マニュアル

省略形……AS.

機能 文字や記号、数字などをキャラクタコードに変換します。

書式

ASC { "文字" / 文字変数 }

参照 CHR$

説明 ●この関数は文字や記号、数字などをキャラクタコードに変換する関数です。たとえば「Z」という文字のキャラクタコードを知りたい場合は

A=ASC"Z"

として実行すれば、Aには「Z」のキャラクタコードが10進数の90として代入されます。

なお、文字が2文字以上指定された場合は、先頭の文字のみがキャラクタコードに変換されます。

〈例〉
```
10 CLS
20 A$=INKEY$
30 IF A$="" THEN 20
40 B=ASC A$
50 PRINT A$;"␣␣";B
```
プログラムをスタートさせたあと、アルファベットや数字キーなどを押せば、その文字のキャラクタコードを表示します。

●文字、数字、記号など（これらを総称してキャラクタといいます）を計算機が記憶したり、処理したりする場合は、すべて計算機が取り扱いやすい数値に変換します。たとえばアルファベットのAは計算機内では65（10進数）という数値（コード）になっています。（実際には2進数の01000001となっています。）同様にBは66、Cは67というようにコードを決めています。このコードの決めかたに何種類かあって、代表的なものにアスキーコード（ASCII code）とJISコードがあります。
本機ではJISコードを元にして作成されたキャラクタとそのコードを使用しています。（266ページのキャラクタ・コード表を参照してください。）

基本命令

3 BEEP………ビープ

省略形……B.

プログラム　マニュアル

機能　音(ピー音)を発生させます。

書式　BEEP　式$_1$[,［式$_2$][, 式$_3$]]

説明　本機は音発生機能を持っており、BEEP命令に続く式の値で音の制御を行うことができます。

式$_1$：音の発生回数を指定します。式$_1$の値は0～65535の範囲内にしてください。

式$_2$：音の高さを指定します。式$_2$の値を255～0にすることにより、約230Hz～約8kHzの周波数の音を指定することができます。

式$_3$：音の長さを指定します。式$_3$の値は0～65535の範囲内で指定してください。
なお、音の長さは式$_2$の値(周波数)により大きく変化します。式$_2$の値が大きくなる（周波数が低くなる)と、式$_3$の値が同じでも音が出ている時間が長くなります。

●式$_3$を省略した場合は、その値が1400に指定され、式$_2$の値を省略した場合は音の高さ(周波数)が約4kHzに指定されます。

〈例〉 BEEP10, 15, 400　　指定された音(ピー音)を10回鳴らします。

参考　BEEP命令を用いて音階を作る場合の目安になるおおよその式$_2$の値を示します。

	ド	ド#	レ	レ#	ミ	ファ	ファ#	ソ	ソ#	ラ	ラ#	シ
高	7											
↑	21		18		15	14		12		10		8
	49	46	43	40	37	35		30		26		22
	105	99	93	87	82	77	72	68	64	60	56	52
↓	219	206	194	182	172	162	152	143	135	127	119	112
低											246	232

なお、BEEP　A，B，C　において、

音の回数は　A回

音の高さは　1300000／(166＋22＊B)Hz

音の長さは　C＊(166＋22＊B)／1300000秒

になります。

基本命令

4　CALL………コール

省略形……CA.

プログラム　マニュアル

機能　機械語プログラムを実行します。

書式　CALL　番地

参照　PEEK、POKE

説明　指定された番地を機械語プログラム実行開始番地として機械語プログラムの実行を開始します。

CALL　&H1F58

（注）この命令を誤って使用するとBASICプログラムや本機のシステムエリアを破壊し、異常が発生することがあります。この場合はリセットスイッチ❺を押して、メモリの内容を消去してください。

関数

5　CHR$………キャラクタドル

省略形……CH.

プログラム　マニュアル

機能　キャラクタコードを文字や記号など（キャラクタ）に変換します。

書式　CHR$　式

参照　ASC

説明　CHR$はASC命令とは逆の関数で、キャラクタコードを文字や記号、数字（これらを総称してキャラクタといいます）に変換します。たとえば、キャラクタコード「90」の文字を知りたい場合は

A$=CHR$9Ø

として実行すれば、A$には"Z"が代入されます。

●本機で扱うことのできるキャラクタと、それに対応するコードは266ページの「キャラクタ・コード表」をご覧ください。

〈例〉
```
1Ø  AA$=""
2Ø  INPUT"コード=";A:CLS
3Ø  AA$=AA$+CHR$A
4Ø  LOCATE  7,1:PRINT  AA$
5Ø  GOTO  2Ø
```

このプログラムは、キャラクタコードを入力し、それを文字や記号に変換して文字変数AA$に順次格納していきます。変換されたキャラクタはそのつど表示されます。

プログラムをスタートさせ、次のコードを入れてみてください。

233、 177、 183、 186

256以上の値が指定されたときは、エラー33になります。

基本命令

6 CLEAR……クリア

プログラム マニュアル

省略形……CL.

機能 固定変数の内容および配列変数、単純変数を消去します。

書式 CLEAR

参照 DIM

説明 単純変数や、DIM命令により確保されていた配列変数はすべて消去（未定義の状態に）され、固定変数の内容もすべて消されます。

〈例〉 1ØØ CLEAR：DIM B(4)

（この例では、すべての変数を消去してから、配列変数 B（4）の定義を行っています。プログラムの先頭ではよくこのような方法を用います。）

カセット命令

7 CLOAD……カセット・ロード

マニュアル

省略形……CLO.

機能 テープから計算機にプログラムを読み込ませます。（マニュアル操作でのみ有効）

書式 （1）CLOAD "ファイル名" ↵

（2）CLOAD ↵

参照 CLOAD?、CSAVE、INPUT#

説明 テープに記録されているプログラムを計算機に読み込むための命令です。

PROモードまたはRUNモードで実行します。

●書式（1）では、指定したファイル名を検索して、そのファイル名のついているプログラムを計算機に読み込みます。

〈例〉 PROモードかRUNモードを指定

CLOAD "PRO1" ↵

（この例では、ファイル名が「PRO1」のプログラムを検索して、これを計算機に読み込みます。）

●書式（2）では、テープが回り始めて最初に記録されているプログラムを計算機に読み込みます。

●読み込みが行われているときは、画面右下に＊マークが表示されます。読み込みが終われば、＊マークは消えます。

ファイル名を検索しているときは、まだ読み込みが行われていませんので＊マークは表示されません。

これは、CLOAD?およびINPUT#命令でも同じです。

なお、読み込みが終われば、プロンプト記号“＞”が表示されます。

●指定されたファイル名が検索できなかった場合は、テープが回り終わっても計算機はファイル名を検索し続けます。この場合は BREAK ON キーを押して検索を止めてください。これは、CLOAD?およびINPUT#命令でも同じです。
●CLOAD命令を実行中にエラーが発生すると、計算機内のプログラムが無効になります。
●本命令を実行するとオープンされていた全ファイルが閉じられます。(クローズされます。)

カセット命令

8 CLOAD M…カセット・ロード・エム

マニュアル

省略形……CLO. M

機能 テープから機械語プログラムを読み込みます。

書式 (1) CLOAD M [ロード番地] ↵
(2) CLOAD M"ファイル名" [;ロード番地] ↵

参照 CLOAD、CSAVE M

説明 ●CSAVE M命令でテープに書き込んだ機械語プログラムを読み込みます。そのとき、本体のプログラムは消去しません。
●「ロード番地」を省略すると、CSAVE M命令で指定した番地に読み込みます。
●「ロード番地」を指定したときは、指定した番地を開始番地として読み込みます。
●ファイル名を省略したときは、テープが回り始めて最初に記録されているプログラムを読み込みます。

カセット命令

9 CLOAD?…カセット・ロード?

マニュアル

省略形……CLO. ?

機能 計算機内のプログラムとテープに記録されている内容との照合を行います。(マニュアル操作でのみ有効)

書式 (1) CLOAD?"ファイル名"↵
(2) CLOAD?↵

参照 CLOAD、CSAVE

説明 照合は、プログラムが正しくテープに記録されたか、あるいはテープから正しく読み込まれたか確認するために行います。
プログラムを照合するときは、PROモードあるいはRUNモードでCLOAD?命令を実行します。
●書式(1)では、指定されたファイル名のついているプログラムと、計算機内のプログラムとの照合を行います。
〈例〉 PROモードかRUNモードを指定
CLOAD?"PRO1"↵
(ファイル名が「PRO1」のプログラムを検索し、計算機のプログラムと照合します。)
●書式(2)では、テープが回り始めて最初に記録されているプログラムと計算機内のプログラムとの照合を行います。

●照合において、もし内容に不一致が生じたときはエラー82になります。

●プログラムの照合が行われているときは画面右下に＊マークが表示されます。
照合が終われば＊マークは消え、プロンプト(>)表示になります。

ファイル関連命令

10 CLOSE……クローズ

省略形……CLOS.

プログラム マニュアル

機能 入出力のファイルを閉じます。

書式 CLOSE［#1］

参照 OPEN、PRINT#

説明 オープン命令で開いたファイルを閉じます。

●SIO（シリアル入出力装置）の回路をソフト的に閉じます。したがって、SIOへの入出力は行われなくなります。

●カセットへのファイル操作を終了します。ファイルが出力（OUTPUT）モードでオープンしていた場合は、メモリ（出力バッファ）に残っているデータおよびファイル終了コードを出力してからクローズ(閉じ)します。

●次の場合も自動的にファイルを閉じます。

- END、NEW、RUN命令を実行したとき
- 電源が切れたとき
- プログラムの編集をしたとき(プログラム入力、修正、削除およびDELETE命令を実行したとき)
- ファイルエリアへのプログラムの登録、呼び出し、削除などを行ったとき
- CSAVE、CLOAD、CLOAD?、CSAVE M、CLOAD M命令実行時
- 機械語モニタモードへ切り替えたとき

（注） BASICモード（RUN、PRO）から他のモード（TEXT、CASL）へ切り替えるときは、クローズをしてから切り替えてください。

基本命令

11 CLS…………クリアスクリーン

省略形……なし

プログラム

機能 表示内容を消去します。

書式 CLS

参照 LOCATE

説明 表示内容を消去し、表示開始ポジションを（Ø，Ø）位置に戻します。

基本命令

12 CONT……コンティニュー

マニュアル

省略形……C.

機能 一時停止しているプログラムの実行を再開します。（RUNモードのマニュアル操作でのみ有効）

書式 CONT ↵

参照 STOP

説明 STOP命令や BREAK ON キーによりプログラムが一時停止しているとき、実行を再開させます。

カセット命令

13 CSAVE……カセット・セーブ

プログラム
マニュアル

省略形……CS.

機能 プログラムをテープに記録します。

書式 (1) CSAVE"ファイル名"
(2) CSAVE
(3) CSAVE"ファイル名"，"パスワード"
(4) CSAVE，"パスワード"

参照 PASS、CLOAD、CLOAD?、PRINT#

説明 ●書式（1）の形では、指定したファイル名をつけて、テープに記録します。
●書式（2）の形では、ファイル名なしで、テープに記録します。
●書式（3）、（4）の形では、プログラムがパスワードつきのプログラムとして記録され、CLOAD命令により計算機に読み込まれたときは、秘密プログラムとなります。
したがって、解除するにはPASS命令でもう一度同じパスワードを宣言しなければなりません。
●パスワードは最大8文字までの英文字、数字、英記号、カナ文字、カナ記号などを用いることができます。" "（Null）をパスワードにすることはできません。
●計算機内のプログラムが秘密化されているときは、CSAVE命令が無視されます。
●本命令を実行するとオープンしていた全ファイルが閉じられます。

◎ファイル名

ファイル名はプログラムなどにつける名前（見出し）のことで、ファイル名をつけてプログラムを記録しておけば、計算機によりファイル名を探し出して読み込みなどを行うことができるので便利です。

●ファイル名は最大8文字までの英文字、数字、英記号、カナ文字、カナ記号などを用いることができます。

●同じテープの同じ面（A面あるいはB面）に、同じファイル名で内容の違うプログラムを記録することは避けてください。同じファイル名のものが2つ以上ありますと、テープから読み込む（転送する）ときにまちがった内容を読み込む恐れがあります。

カセット命令

14 CSAVE M…カセット・セーブ・エム

プログラム マニュアル

省略……… CS. M

機能 機械語プログラムをテープに記録します。

書式 CSAVE M ["ファイル名";] 開始番地，終了番地

参照 CSAVE、CLOAD M

説明 ●機械語プログラムにファイル名をつけてバイナリ形式でテープに記録します。
ファイル名を省略したときは、ファイル名なしで記録します。
●指定した開始番地から終了番地までの内容を書き込みます。

基本命令

15 DATA………データ

プログラム

省略形……… DA.

機能 READ文に続く変数に与えるデータを指定します。

書式 DATA {式 / 文字列}, {式 / 文字列}……

参照 READ

説明 READ命令の説明を参照ください。(235ページ参照)

基本命令

16 DEGREE…ディグリー

プログラム マニュアル

省略形…… DE.

機能 角度単位を“度”に設定します。

書式 DEGREE

参照 RADIAN、GRAD

説明 三角関数、逆三角関数、座標変換関数で扱う角度の単位を“度”単位〔°〕に設定します。(1直角=90°)

17 DELETE…デリート

マニュアル

省略形……DEL.

機能 プログラムの行を削除します。（PROモードのマニュアル操作でのみ有効）

書式 （1）DELETE　開始行番号［－［終了行番号］］↵

（2）DELETE　－終了行番号↵

参照 NEW、PASS

説明
●開始行番号と終了行番号を指定したときは、その間に含まれるすべての行を削除します。
●開始行番号のみを指定したときは、その行だけを削除します。
●開始行番号とハイフン（－）を指定したときは、開始行番号以上の番号を持つ行をすべて削除します。
●書式（2）の場合は、プログラムの先頭行から終了行番号間に含まれるすべての行を削除します。
●開始行番号と終了行番号の両方を省略したときは、エラー10になります。
●指定した行番号が存在しないときは、エラー40になります。
●開始行番号が終了行番号よりも大きい指定をすると、エラー44になります。
●パスワードが設定されているときは、DELETE命令は無視されます。

18 DIM…………ディメンジョン

プログラム　マニュアル

省略形……D.

機能 配列名と、その大きさを定義（宣言）し、配列変数をメモリ（プログラム・データエリア）上に確保します。

書式

（1）DIM {配列名（式₁） / 配列名（式₁，式₂）} [, {配列名（式₁） / 配列名（式₁，式₂）}]…

（2）DIM {配列名（式₁） / 配列名（式₁，式₂）} ＊式₃ [, {配列名（式₁） / 配列名（式₁，式₂）} ＊式₃…]

（注）書式（2）は文字変数でのみ使用できます。

参照 CLEAR、RUN

説明
●配列変数を使用するときは、事前にDIM命令により配列名と、その大きさを定義（宣言）して、メモリ（プログラム・データエリア）上に確保しておく必要があります。
●配列名はアルファベット1文字あるいは2文字（2文字目は数字も使用可能）で指定し、文字配列変数の場合は後に＄マークを付けます。
●式₁および式₂は添字といわれ、配列の大きさ(配列要素数)と次元を指定します。添字が1つのものを一次元配列と呼び、2つのものを二次元配列と呼びます。

〈例〉 DIM　B（3）　　一次元配列変数B（　）について、配列要素B（Ø）、B（1）、B（2）、B（3）の4個が確保されます。

DIM　XA＄（2，3）二次元配列変数XA＄(　)について、配列要素XA＄(Ø，Ø)、XA＄(Ø，1)、………XA＄（2，2）、XA＄（2，3）の12個が確保されます。

添字は理論的にØ～255までの整数値を用いることができますが、計算機のメモリの大きさ、使用状態

によっては添字で指定しただけ、変数が確保できない場合があります。（確保できないときはエラー60になります。）

●添字が小数部を含んでいるときは小数部は無視され、整数部のみが有効になります。

X(2.6)　　→　X（2）とみなします。

Y(0.25)　　→　Y（Ø）とみなします。

●添字は数値変数や式の形で用いることもできます。

〈例〉 1Ø　INPUT　A, B

　　　2Ø　DIM　X（A), Y（B−1）

●文字配列変数は変数の長さを指定することができます。

書式（2）において、式₃により文字数を1～255文字の範囲で任意に指定することができます。

〈例〉 DIM　F$（2）*3Ø　　F$（Ø）～F$（2）の各変数には、それぞれ最大30文字まで記憶することができます。

　　　DIM　Y$（5, 4）*6　　Y$（Ø, Ø）～Y$（5, 4）の各変数には、それぞれ最大6文字まで記憶することができます。

なお、文字数の指定（*式₃）を省略した場合は、自動的に16文字が指定されます。

●複数の配列を使用する場合は、DIM命令で一度に定義することができます。

〈例〉 DIM　J（5), K$（4, 3), XB$（5）*1Ø

●配列変数はCLEAR命令により消去する（未定義の状態にする）ことができます。

また、RUN命令によりプログラムの実行を開始したときも、以前に定義されていた配列変数はすべて消去されます。

●一度定義した配列名を再定義することはできません。

たとえば、DIM　X（5）とDIM　X（3, 4）は同じXという配列名になりますので、同時に使用することはできません。

また、RUN命令によりプログラムを実行したときは、それまでメモリ上に確保されていた配列変数は消去されますが、GOTO命令によるプログラムの実行では、変数は消去されません。

したがって、一度実行したプログラムをGOTO命令により再実行させるときなどに、DIM命令の書かれている行を実行させると同じ変数名を再定義することになり、エラー30になります。

このような場合はCLEAR命令で消去してから、定義し直すようにしてください。

〈例〉 5Ø　"S"：CLEAR：　DIM　X（3, 4）

　　　　⋮

なお、数値配列変数と文字配列変数は別べつの配列とみなされますので、たとえば、配列Z（　）とZ$（　）は同時に使用することができます。

基本命令

19　END…………エンド　　プログラム

省略形……E.

機能　プログラムの実行を終了させます。

書式　END

説明　END命令が実行されますと、プログラムの実行を終了します。

●オープンされていたファイルはすべて閉じられます。

ファイル関連命令

20 FILES……ファイルズ

省略形……FI.

マニュアル

機能 ファイルエリアに登録されているファイルのファイル名を表示します。

書式 FILES ↵

参照 LFILES、SAVE、LOAD

説明 ファイルエリアに登録されているプログラムファイルのファイル名を表示します。

●本命令を実行すると先頭のファイル名から表示します。画面には最大4つのファイル名まで表示されます。

ファイル名の左側に➡マークが点灯しています。➡マークは ↑ 、 ↓ キーで上下に移動できますので、これを移動させていけば必要に応じて画面が送られて、別のファイル名を表示します。

CLS 、 BREAK キー等でこの状態を解除することができます。

➡マークを必要なプログラムのファイル名に移動させてから SHIFT ＋ M(LOAD) キーを押すと、そのプログラムを呼び出すことができます。ただし、TEXTモードで登録したプログラムを呼び出すことはできません。

基本命令

21 FOR～NEXT…フォー～ネクスト

省略形……F. N. STE.

プログラム

機能 FORとNEXTの間に書かれた命令を指定された条件が満たされるまで、くり返し実行します。

書式 FOR　数値変数＝初期値　TO　最終値　STEP　きざみ値

～

NEXT［数値変数］

説明 数値変数が初期値から始まって、指定されたきざみ値分ずつ増加（あるいは減少）していき、最終値よりも大きく（あるいは小さく）なるまでFORとNEXTの間をくり返し実行します。（このくり返し部分をFOR～NEXTループと呼びます。）

```
FOR A=Ø TO 1Ø STEP 2
   ～
NEXT A
```

AがØから始まって、1回FORとNEXT間を実行するごとにAに2を加えながら、Aの値が10を超えるまで、FORとNEXTの間を実行します。

なお、きざみ値が1のときはSTEP　1を省略することができます。

●FORとNEXTは必ず対にして使い、FORの後の数値変数とNEXTの後の数値変数は同一でなければなりません。ただし、NEXTの後の数値変数は省略できます。

```
FOR B=1 TO 5
   ～
NEXT B
```

同じ数値変数にする。

●初期値、最終値、きざみ値（ステップ値）は次の範囲内で指定できます。

－9.999999999E99～9.999999999E99

（$-9.999999999\times10^{99}$～$9.999999999\times10^{99}$）

ただし、初期値にきざみ値を加えると、最終値から離れてしまう場合はループ内を1回だけ実行してループを抜けます。

なお、きざみ値をØに指定しますと、永遠にループ内の実行をくり返すプログラムになってしまいます。

●FOR〜NEXTループの中に、別のFOR〜NEXTループを入れる場合、中に入るFOR〜NEXTループは外のFOR〜NEXTループ内に完全に入っていなければなりません。この条件でループを5段まで重ねて使う（深みをもたせる）ことができます。

●FOR〜NEXTループの外からループ内に飛び込むことはできません。（飛び込ませるとエラー52になります。）

（注1）FOR〜NEXTループから外に飛び出した場合、そのループは終了したことになりませんので、プログラムによっては（FOR命令を何回か実行するようなプログラムの場合）、FOR〜NEXTの深みエラー50が発生することがありますので注意してください。

（注2）FOR〜NEXTループ内ではCLEAR命令は使用できません。

関数

22 FRE…………フリー

省略形……FR.

プログラム　マニュアル

機能　本機内のメモリの中で、未使用部分のバイト数を求めます。

書式　FRE

説明　本機内のメモリの中で、現在使用されていない部分のバイト数が得られます。
（BASICのプログラムや配列変数、単純変数および機械語エリア、ファイルエリア、テキストエリアとして使用されている部分以外のバイト数）

メモリ（30435バイト）

（注）PROモードで使用する場合はPRINT　FRE ↵ とキー操作してください。

基本命令

23 GOSUB~RETURN…ゴーサブ~リターン

プログラム

省略形……GOS. RE.

機能 指定した行から始まるサブルーチンへプログラムの実行を移し、RETURN命令でもどります。

書式
GOSUB {行番号 / ラベル}

RETURN

参照 GOTO、ON GOSUB

説明 ある一連のプログラムで、何回も同じ計算や処理が出てくる場合、その部分を抜き出してプログラムしておき、必要に応じてその抜き出したプログラムを実行すればプログラムを短かく、また簡略化することができます。

●サブルーチンへのジャンプはGOSUB命令に続いて、サブルーチンのおかれている行番号またはラベルを書いて指示します。（ラベルについては143ページ参照）

〈例〉

```
        :
50:GOSUB 200          200行へサブルーチンジャンプ
        :
100:GOSUB "A"         ラベル"A"または＊Aが書かれている行へサブルーチンジャンプ
        :
```

●サブルーチンの最後にはRETURN命令を書いて、メインルーチンへの復帰を指示します。（RETURN命令には行番号の指定は不要です。）

メインルーチンへ復帰したときは、GOSUBのあった次の命令を引き続き実行します。

●サブルーチンから別のサブルーチンへ実行を移すことができます。

サブルーチンからサブルーチンへ、また次のサブルーチンへ……というように重ねて使用する場合、最高10段まで重ねる（深みを持たせる）ことができます。（10段を超えるとエラー50になります。）

基本命令

24 GOTO………ゴートゥー

プログラム
マニュアル

省略形……G.

機能 プログラムの実行を指定された行へ無条件に移します。

書式
GOTO {行番号 / ラベル}

参照 GOSUB、ON GOTO、RUN

説明 通常プログラムは若い行から順次実行されますが、GOTO命令により、その実行を指定した行へ移す（ジャンプさせる）ことができます。

また、RUNモードでのマニュアル操作により、指定した行からプログラムの実行を開始させることができます。

●ジャンプ先は、GOTO命令に続いて行番号またはラベルを書くことによって指定します。(ラベルについては143ページを参照)

〈例〉

GOTO 4Ø　　40行へジャンプしなさい。

GOTO"AB"　　ラベル"AB"または＊ABのついている行へジャンプしなさい。

●指定した行番号およびラベルがない場合はエラー40になります。

●同じラベルが2個以上書かれているときは、行番号の小さいほうへジャンプします。

●GOTO命令によりプログラムの実行が開始されたときの計算機の状態については142ページを参照してください。

基本命令

25 GRAD………グラード

省略形……GR.

プログラム　マニュアル

機能　角度単位を"GRAD"(グラード)に設定します。

書式　GRAD

参照　DEGREE、RADIAN

説明　三角関数、逆三角関数、座標変換関数で扱う角度の単位を"GRAD"単位[^{g}]に設定します。(1直角=100^g)

基本命令

26 IF～THEN…イフ～ゼン

省略形……IF T.

プログラム

機能　条件を判断し、プログラムの流れ(実行の順番)を変えます。

書式　IF 条件式 THEN 実行文

(THENは省略可)

参照　AND、OR

説明　●IFに続く条件が成立した場合はTHENに続く命令が実行されます。条件式が成立しない場合は、THENに続く命令を無視して、次の行へ実行が移されます。

●THENに続いて行番号またはラベルを書くと、その行またはラベルが書かれている行へジャンプします。

●実行文として代入文を書くときは、THENまたはLET命令が必要です。

●THEN命令を省略して実行文を書くことができますが、ジャンプを指示する場合はTHENまたはGOTO命令が必要です。

〈例〉 IF A<5 THEN C=A＊B:GOTO 5Ø

もしAが5よりも小さければ、A＊Bの結果をCに代入して、50行へジャンプします。

IF B=C+1 GOTO 6Ø　　もし、BとC+1の結果が等しければ60行へジャンプします。

または IF B=C+1 THEN 6Ø

●IFに続くA<5などの式を条件式といいます。条件式は通常次のような形で書きます。

条件式	判　断　内　容	
○○ = ××	等しいかどうか判断	(○○は××に等しいか?)
○○ > ××	大きいかどうか判断	(○○は××より大きいか?)
○○>=××	以上かどうか判断	(○○は××以上か?)
○○ < ××	小さいかどうか判断	(○○は××より小さいか?)
○○<=××	以下かどうか判断	(○○は××以下か?)
○○<>××	等しくないかどうか判断	(○○と××は等しくないか?)

(注) ○○および××は式、変数を表します。(5*4、A、8、など)

●条件式をAやB+4のように変数や、通常の式のみにすることもできます。

```
IF B+4 THEN 6Ø
IF A THEN 6Ø
```

この場合は、変数や式の値がØ以外(IF文が成立)ならばIF文の後に続く内容を実行し、Ø(IF文が不成立)ならば次の行を実行します。

●条件式は次の例のように"*"と"+"を用いて結合することができます。

〈例〉 IF (A>5) * (B>1) THEN…………

Aが5よりも大きく、かつBが1よりも大きいとき、THENに続く内容を実行(論理積:AND)

IF (A>5) + (B>1) THEN…………

Aが5よりも大きいか、あるいはBが1よりも大きいとき、THENに続く内容を実行(論理和:OR)

●文字列の比較

条件式に文字列を用いることにより、文字列の比較・大小判断を行うことができます。

文字列はキャラクタコードの大きさによって比較されます。たとえば、キャラクタコードでは"A"が65、"B"が66、"C"が67……となっています。したがって"A"は"B"よりも小さく"B"は"C"よりも小さくなります。同様に数字はアルファベットより小さくなります。

〈例〉 150行のデータを読み込んで、アイウエオ順に並べ替えます。

```
  5 CLS
 1Ø DIM Y$(4)
 2Ø READ Y$(1),Y$(2),Y$(3),Y$(4)
 3Ø GOSUB 2ØØ:PRINT
 4Ø FOR A=1 TO 3
 5Ø FOR B=A+1 TO 4
 6Ø IF Y$(A)<=Y$(B) THEN 8Ø
 7Ø Y$(Ø)=Y$(A):Y$(A)=Y$(B):
    Y$(B)=Y$(Ø)
 8Ø NEXT B:NEXT A
 9Ø GOSUB 2ØØ:END
15Ø DATA アサミ,サチコ,アサコ,キヨミ
2ØØ FOR A=1 TO 4
21Ø PRINT Y$(A);"␣␣";
22Ø NEXT A
23Ø PRINT:RETURN
```

(40〜80行: 比較、並べ替えループ)

●式の中での文字列の長さについて

文字列の結合や大小比較などの式において、その中に含まれる文字（アルファベット、カタカナ、数字など）の数は、合計で 255 文字を超えない範囲で演算処理を行ってください。

関数

27 INKEY$…インキードル

省略形…… INK.

プログラム

機能 INKEY$命令が実行されたとき、いずれかのキーが押されていれば、その内容を読み込んで指定された文字変数に代入します。

書式 文字変数＝INKEY$

説明 INKEY$命令は通常、次の例のようにくり返しループを作って、有効なキーが押されるのを待ちます。

〈例〉

```
10 CLS
20 Z$=INKEY$
30 IF Z$=""THEN 20
40 PRINT"---";ASC Z$;"---"
50 Z$=INKEY$
60 IF Z$=""THEN 10
70 GOTO 50
```

（20〜30：このラインをくり返し実行し、キーが押されるのを待ちます。）

● INKEY$命令が実行されたときに、キーが押されていなければ、変数にはNull（空白）が代入されます。

● BREAK ON キーはプログラムの一時停止キー（ブレークキー）として働きます。

● INKEY$命令は SHIFT キーを押しながら、キーを押したときに働く機能や入力される記号、 2ndF キーに続いて押したときに働く機能などを読み込むことはできません。また、英小文字やカナ文字を読み込むことはできません。

● OFF 、 SHIFT キーはINKEY$命令で読み込むことはできません。

●プログラムのはじめにINKEY$がありますとプログラムをスタートさせたとき、スタートキーを読み取ってしまうことがありますので、注意が必要です。

● INKEY$命令は、この命令実行時に押されていたキーのデータを読み込むための命令ですので、INPUT命令のようにキー入力時に ↵ キーを押す必要はありません。

● INKEY$命令は、記号、数字、英大文字などのような印字可能文字の他に、次の表のØ列および1列に示す各種の制御キー（非印字文字）も読み取ることができます。

このような非印字文字はPRINT文では表示できないので（すべてスペースで表示されます）、表示させたい場合は上記のプログラム例のようにASC命令でアスキーコードに変換するとよいでしょう。

INKEY$で読み取られるキーとキーコード

		0	16	32	48	64	80	…	128	144	…	240
	16進上位 / 16進下位	0	1	2	3	4	5	…	8	9	…	F
0	0		2nd F	SPACE	0		P					
1	1				1	A	Q			ln		
2	2	CLS			2	B	R			log		
3	3				3	C	S					
4	4	↑	カナ		4	D	T					
5	5	↓	CAPS		5	E	U			sin		
6	6				6	F	V			cos		
7	7	ANS	BS		7	G	W		$1/x$	tan		
8	8	BASIC	R·CM	(	8	H	X		x^2			
9	9	TEXT	M+	)	9	I	Y					
10	A	TAB		*		J	Z					
11	B	INS		+	;	K				→DEG		π
12	C	CONST		,		L				F↔E		$\sqrt{\ }$
13	D	↵		−		M				$_nP_r$		
14	E	►		·		N	y^x ∧			MDF		
15	F	◄	Exp	/		O						

基本命令

28 INPUT……インプット

プログラム

省略形……I.

機能 プログラムの実行を一時止め、マニュアル操作により数値や文字を変数に代入するための命令です。

書式 （1）INPUT　変数［，変数……］

（2）INPUT "文字"，変数［，"文字"，変数……］

（3）INPUT "文字"；変数［，"文字"；変数……］

参照 INPUT#、INKEY$、READ、LOCATE

説明 プログラム実行において、計算機にデータを与えるため、プログラムを一時停止させる命令で、INPUT命令に続いて、データを格納するための変数を指定します。

●書式（1）の形の命令が実行されますと、計算機はデータ入力待ちのため、?を表示してプログラムの実行を停止します。このとき、キーを操作してデータを入れ、↵キーを押せば、データが変数に代入されて、実行が再開されます。

（変数を複数個指定する場合は，（コンマ）で区切って指定します。）

●書式（2）の形では、データ入力待ちのとき"　"で囲まれた文字を入力ガイダンス（入力案内）として表示します。

この形の場合、データを入力しますと入力ガイダンスは消えます。

［入力ガイダンスは、計算機がデータ待ちになったとき、何のデータを要求しているのか、また、どのデータを入力すればよいか、などをわかりやすくする目的で、メッセージを表示させるものです。］

●書式（3）の形では、書式（2）の場合と同じくデータ入力待ちのとき、入力ガイダンスが表示されますが、データを入力したとき、その入力ガイダンスに続けて、入力したデータが表示されます。

●書式（1）、（2）、（3）は混合して用いることができます。

〈例〉 INPUT "A=";A, B, "C=?", C

●INPUT命令の入力待ちのときに、データを入力せずに ↵ キーのみを押したときは、変数に入っていたデータを保持したまま、次の実行に移ります。

●INPUT命令に続いて指定された変数の型と、入力するデータの型は同じでなければなりません。特に、数字を入力する場合は、数値として入力するのか、あるいは文字として入力するのかの区別に注意してください。

（文字変数には文字データ、数値変数には数値データを入力します。）

●INPUT命令実行前に、LOCATE命令により表示開始位置が指定されている場合は、その位置から入力ガイダンス、あるいは?が表示されます。

（注） INPUT命令による入力時のエラーは CLS キーで解除して、正しいデータを入れてください。

ファイル関連命令

29 INPUT#…インプット・クロスハッチ

プログラム

省略形…… I. #

機能 ファイルのデータを指定した変数に代入します。

書式 INPUT #ファイル番号, 変数 [, 変数…]

参照 OPEN、PRINT#

説明 SIO（シリアル入出力装置）に送られてくるデータ、またはカセットテープに記録されているデータを指定されている変数に代入します。

●OPEN命令で"COM:"を指定、または"CAS:"のINPUTを指定してオープンしているときにのみ有効です。

●ファイル番号には1を指定します。

●"変数"には、次の変数が指定できます。

固定変数	A、X、B$など
単純変数	AB、CD、EF$など
配列の要素	B（1Ø）、C$（5，5）など
配列全体	B（*）、C$（*）など

●指定した変数の数よりもファイルのデータが少ない場合は、エラー87になります。

〔変数への読み込み規則〕

●数値

・データの前にスペースがあった場合は無視します。スペースでない最初の文字をデータの始まりとします。

・データの区切りは、コンマ（，）およびスペース、CR（&HØD）、LF（&HØA）コードです。

・数値化できないデータを読み込んだ場合はØとします。

●文字

・データの前にスペースがあった場合は無視します。スペースでない最初の文字をデータの始まりとします。

●データの始まりがダブルクォーテーション（"）のときは、次のダブルクォーテーションまでを1つのデータとします。

●データの始まりがダブルクォーテーションでないときは、データの区切りをコンマ（，）およびCR（&HØD）、LF（&HØA）コードを読み込んだとき、あるいは256文字目を読み込んだときとします。

●EOF（エンドオブファイル：&H1A）コードの処理

●データの前にEOFコードがある場合はエラーになります。

●データの途中にEOFコードがある場合はデータの区切りと見なします。

●配列全体への入力順序

〈例〉 一次元配列の場合　B（Ø）→B（1）→B（2）→……

二次元配列の場合　C（Ø，Ø）→C（Ø，1）→C（Ø，2）→……

C（1，Ø）→C（1，1）→C（1，2）→……

⋮

ファイル関連命令

30 KILL………キル

省略形……K.

マニュアル

機能　ファイルを消去します。

書式　KILL"ファイル名" ↵

説明　●ファイルエリア内の指定したファイルを消去します。

指定したファイルが存在しないときはエラー94になります。

関数

31 LEFT$……レフトドル

省略形……LEF.

プログラム　マニュアル

機能　文字列の左側から何文字かを取り出します。

書式　LEFT$（文字列，式）

参照　MID$、RIGHT$

説明　指定された文字列の左から、式の値で指定された桁数（文字数）だけ、文字を取り出します。たとえばA$="ABCDE"のときLEFT$（A$，3）はA$の文字列の左側3文字、すなわち"ABC"を取り出しなさいという意味になります。

●式の値は、Ø～255の範囲の整数でなければなりません。

関数

32 LEN…………レングス

プログラム　マニュアル

省略形……なし

機能　文字列の文字数を求めます。

書式　LEN　文字列

説明　1つの文字列の中に含まれる文字の数（記号、スペース、数字も含みます）を求める関数です。

たとえば

A=LEN"ABC1234バン"

のようにして実行すれば、"ABC1234バン"の文字数10が変数Aに代入されます。（だく点や半だく点も1文字と数えられます。）

基本命令

33 LET…………レット

プログラム

省略形……LE.

機能　変数に数値や文字を代入するための命令です。

書式　［LET］変数=データ［，変数=データ］…

説明　代入文はLET命令に続いて代入式を書きます。

代入式はA=5+3、B$="ABC"のように、左辺に変数を、右辺に式や文字列を書きます。この場合の“=”は“等しい”という意味ではなく、“左辺の変数に、右辺の内容あるいは計算結果を入れなさい”という意味です。

●代入文はLETを省略して使用できます。

●変数とデータの型は同じ（文字型どうし、または数値型どうし）でなければいけません。

ファイル関連命令

34 LFILES…エルファイルズ

マニュアル

省略形……LF.

機能　ファイルエリアに登録されているファイルのファイル名を印字します。

書式　LFILES ↵

参照　FILES

説明　●ファイルエリアに登録されているすべてのファイル名をプリンタで印字します。

35 LIST………リスト

マニュアル

省略形……L.

機能 記憶されているプログラムを表示させます。（PROモードのマニュアル操作でのみ有効）

書式 （1）LIST ↵

（2）LIST 行番号 ↵

（3）LIST ラベル ↵

参照 LLIST、PASS

説明 表示部にプログラムを表示させます。

●書式（1）では、プログラムの先頭行から、表示できる範囲で表示します。

●書式（2）では、指定した行番号の行から、表示できる範囲で表示します。もし、指定した番号の行がない場合は、それよりも大きく、かつ一番近い行から表示します。

●書式（3）では、指定したラベルの書かれている行から、表示できる範囲で表示します。ラベルは143ページを参照ください。

●プログラムが記憶されていない場合は、LIST命令は無視されます。

●パスワードが設定されているときは、LIST命令は無視されます。

●プログラム内にないラベルや、プログラムの最終行よりも大きい行番号を指定した場合はエラー40になります。

プリンタ命令

36 LLIST……ラインリスト

マニュアル

省略形……LL.

機能 プログラムをプリンタで書き出します。（PROおよびRUNモードのマニュアル操作でのみ有効）

書式 （1）LLIST ↵

（2）LLIST {行番号 / ラベル} ↵

（3）LLIST［開始行］－［終了行］↵

参照 LIST、PASS

説明 ●書式（1）では、計算機内のプログラムをすべて書き出します。

●書式（2）では、指定した行番号または指定したラベルのついている行を書き出します。

●書式（3）では、指定した開始行から終了行までのプログラムを書き出します。（開始行 および終了行はラベルも可）

なお、書式（3）では開始行または終了行の指定を省略することができます。（同時に両方を省略することはできません。）

開始行を省略した場合は、プログラムの先頭行から、指定した終了行までのプログラムを書き出します。

終了行を省略した場合は、指定した開始行から最終行までのプログラムを書き出します。

〈例〉 LLIST-200 ↵　　先頭から200行まで書き出します。

LLIST100- ↵　　100行から最後の行まで書き出します。

●指定した番号の行がない場合は、それぞれの値よりも大きくかつ最も近い行が指定されます。ただし、開始行が終了行よりも大きくなるような指定をするとエラー44になります。

●パスワードが設定されているときは、LLIST命令は無視されます。

ファイル関連命令

37 LOAD………ロード

マニュアル

省略形……LO.

機能　ファイルエリアに登録されているBASICプログラムを呼び出します。

書式　LOAD"ファイル名" ↵

参照　SAVE

説明　●ファイルエリアから、指定したファイル名のプログラムを呼び出します。

●拡張子が「.BAS」のときのみ、拡張子の記述は省略できます。

●読み込み終了時、SIOやカセットがオープンしていると、それらはクローズされます。

●テキスト(TEXT)プログラムを呼び出そうとするとエラー96になります。

基本命令

38 LOCATE…ロケート

プログラム

省略形……LOC.

機能　表示の開始位置(ポジション)を指定します。

書式　LOCATE　式1[,式2]

参照　CLS、INPUT、PRINT

説明　PRINT命令などで表示される内容の表示開始位置(カーソルの位置)を指定します。

●表示位置は、次の図のようになります。

横の位置(式1で指定) ──→ 01234567891011121314151617181920212223

0
1
2
3

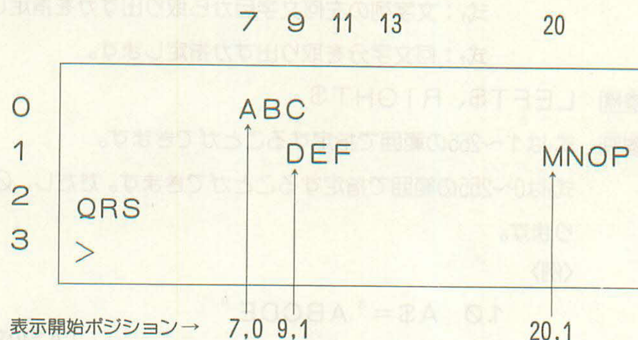
縦の位置(式2で指定) ──→

このように、表示部を横と縦に分け、式1の値で横の位置を指定し、式2の値で縦の位置を指定します。

●式1の値は0~23、式2の値は0~3の範囲で指定します。この範囲外ではエラー33になります。

●式₂を省略した場合、縦の位置は現在カーソルがある位置になります。

〈例〉
```
 5 CLS
1Ø LOCATE 7, Ø:PRINT"ABC"
2Ø LOCATE 9, 1:PRINT"DEF"
3Ø LOCATE 2Ø, 1:PRINT"MNOPQRS"
```
プログラムを実行すると、次のように表示されます。



●LOCATE命令で表示開始ポジションを指定した場合は、表示の一部分だけを変えることもでき、応用範囲も広がります。なお、表示を消去するときはCLS命令を用います。

●LOCATE命令による指定はINPUT命令に対しても働きます。

プリンタ命令

39 LPRINT…ラインプリント

省略形……LP.

プログラム
マニュアル

機能 指定した内容をプリンタで印字させます。

書式 LPRINT [{ 式 / 文字列 }] [{ , / ; } { 式 / 文字列 } ……] [;]]

参照 PRINT、USING

説明 ●項目を1つだけ指定したときは、式の値は紙の右側にツメて印字し、文字は紙の左端から印字します。文字列が24桁を超える場合は自動的に改行して印字します。

●コンマ（，）を入れて2つ以上の項目を記述すると、1行の印字桁数24桁を左右12桁に分けて印字します。このときも12桁の範囲内で数値は右ヅメ、文字は左ヅメにします。なお、印字内容が12桁を超える場合、数値は仮数部の下位桁を切り捨てて12桁以内で印字し、文字は先頭から12桁を印字します。

●区切りにセミコロン（；）を使用している場合は紙の左端から連続的に印字します。印字内容が24桁を超える場合は自動的に改行されます。

●プログラム中で末尾がセミコロン（；）の場合は、印字が終了しても改行を行わず、次のLPRINT命令で、指定されている内容を前の内容に続けて印字します。

●LPRINTのみで、印字する内容が指定されていないときは改行を行います。

関数

40 MID$………ミッドドル

プログラム　マニュアル

省略形……MI.

機能　文字列の中から一部分の文字列を取り出します。

書式　MID$（文字列，式₁，式₂）

式₁：文字列の左何文字目から取り出すかを指定します。

式₂：何文字分を取り出すか指定します。

参照　LEFT$、RIGHT$

説明　式₁は1～255の範囲で指定することができます。

式₂は0～255の範囲で指定することができます。ただし、Øを指定した場合、文字は得られず Null になります。

〈例〉

```
1Ø  A$="ABCDE"
2Ø  B$=MID$(A$, 2, 3)
3Ø  PRINT  B$
```

A$の文字列の左側2文字目から3文字、つまりBCDを取り出し、B$に代入します。

41 MON…………モニタ

マニュアル

省略形……MO.

機能　機械語モニタモードにします。（RUNおよびPROモードのマニュアル操作でのみ有効）

書式　MON↵

説明　機械語モニタモードにします。「機械語モニタ」の説明を参照ください。

基本命令

42 NEW…………ニュー

マニュアル

省略形……なし

機能　プログラムとデータを消去します。（PROモードのマニュアル操作でのみ有効）

書式　NEW↵

参照　CLEAR、PASS

説明　●PROモードでNEW命令を実行しますと、プログラム・データエリア内のプログラム（BASICプログラム）や配列変数、単純変数がすべて消去され、固定変数の内容も消去されます。

●オープンしているファイル（デバイス）をクローズします。

●パスワードが設定されているときはNEW命令が無視されます。

43 NOT…………ノット

プログラム マニュアル

省略形……NO.

機能 与えられた数値の否定を取ります。

書式 NOT　式

参照 AND、OR

説明 2進数において、否定は次の値を取ります。

NOT　1=Ø

NOT　Ø=1

このように2進数においては、1の否定はØになり，Øの否定は1になります。なお、10進数の否定を取った場合も計算機内では、その10進数を2進数に変換して、各桁の否定を取り、その結果を10進数に変換します。

このときの10進数をXとしたとき、Xとその否定（NOT　X）の間には次の関係があります。

NOTX=−（X+1）

この関係式から

NOT　Ø=−1

NOT−1=Ø

NOT−2=1

となります。

44 ON～GOTO……オン～ゴートゥー
ON～GOSUB…オン～ゴーサブ

プログラム

省略形……O. G.、O. GOS.

機能 式の値により、いくつかの指定された行へ選択的に実行を移します。

書式 ON　式 {GOTO / GOSUB} 行番号$_1$［，行番号$_2$］［，行番号$_3$］……

参照 GOTO、GOSUB

説明 ONに続く“式”の値が1であれば“行番号$_1$”、2であれば“行番号$_2$”、3であれば“行番号$_3$”というように、“式”の値により何番目に指定されている“行番号”かが決定され、GOTOやGOSUBの各機能を実行します。

- “式”の値は整数部のみが有効になります。
- “式”の値が1より小さいとき、および指定されている“行番号”の数よりも大きいときは、本命令の次の命令へ実行が移ります。
- “行番号”はラベルを指定することもできます。（ラベルについては143ページ参照）

```
〈例〉 1Ø CLS
      2Ø LOCATE 5,Ø
      3Ø INPUT"バンゴウ(1-3)?",N
      4Ø LOCATE 1Ø,2
      5Ø ON N GOTO 1ØØ,2ØØ,3ØØ
      6Ø CLS:GOTO 2Ø
      7Ø END
      1ØØ PRINT"グウ"
      11Ø GOTO 2Ø
      2ØØ PRINT"チョキ"
      21Ø GOTO 2Ø
      3ØØ PRINT"パア"
      31Ø GOTO 2Ø
```

ファイル関連命令

45 OPEN………オープン

省略形……OP.

プログラム　マニュアル

機能　SIO（シリアル入出力装置）に対するデータの入出力、ミニI／Oの出力、カセットテープに対するデータの入力または出力を可能にします。

書式　(1) OPEN"COM:"
(2) OPEN"CAS:[ファイル名]"FOR OUTPUT
(3) OPEN"CAS:[ファイル名]"FOR INPUT
(4) OPEN"LPRT:"

参照　CLOSE

説明　●書式（1）ではSIOに対する入力、出力を可能にします。（SIO回路をオープンします。）ファイル名を書くことや、入出力の指定はできません。

●書式（2）ではカセットテープに対するデータの出力を可能にします。
この形でOPEN命令を実行すると、OUTPUT（出力）モードで回路（ファイル）をオープンし、インフォメーションブロック（見出し部）をテープに記録します。
ファイル名が指定されている場合は、インフォメーションブロックにファイル名も記録されます。

●書式（3）ではカセットテープからのデータの入力を可能にします。
この形でOPEN命令を実行すると、INPUT（入力）モードで回路（ファイル）をオープンし、指定されたファイル名のあるインフォメーションブロックを検索し、読み込みます。ファイル名が指定されていない場合は、テープが回り始めて最初に記録されているインフォメーションブロックを読み込みます。

（注）指定されたファイル名が検索できなかった場合は、テープが回り終わっても計算機は検索を続けます。この場合は、BREAKキーを押して検索を止めてください。

●書式（4）ではミニI／Oに対する出力を可能にします。（245ページ参照）

●複数の回路（ファイル）を同時にオープンしておくことはできません。
どれか1つがオープンしているときにOPEN命令を実行するとエラー86になります。

関数

46 OR……………オア

プログラム マニュアル

省略形……なし

機能 式と式との論理和を計算します。また、条件式の結合を行います。

書式 式 OR 式

条件式 OR 条件式

参照 AND、NOT、IF

説明 ●2進法において、論理和は次のような値をとります。

1 OR 1=1

1 OR Ø=1

Ø OR 1=1

Ø OR Ø=Ø

なお、10進数の論理和を求めた場合でも、計算機内では10進数を2進数に変換したうえで、各桁の論理和を求め、その結果を10進数にもどします。

たとえば、41と27の論理和は次のように計算されます。

41 OR 27=59

```
   ┌101001………41
OR<
   └011011………27
   ─────────────
    111011………59
```

41と27をそれぞれ2進数に変換し、各桁のORを取ります。そして、その結果を10進数に変換すれば59になります。

●2つ以上の条件のいずれかを満足するような条件を1つの式で表わします。

〈例〉 A<Ø OR A>6　　Aは0よりも小さいか、あるいは6よりも大きい。

IF A=1 OR B=1 OR C=1 THEN…

A、B、Cのいずれかが1のとき、THENに続く命令を実行します。

基本命令

47 PASS………パス

マニュアル

省略形……PA.

機能 パスワードの設定あるいは解除を行います。（PROおよびRUNモードのマニュアル操作でのみ有効）

書式 PASS"文字" ↵

参照 NEW、CLOAD、CSAVE、SAVE

説明 作成したプログラムを他の人に知られたくないときや、プログラムを変更されたくないときに、ある言葉や記号をパスワード（暗号）とし、パスワードが与えられないかぎり、計算機内のプログラムを呼び出すことができないようにすることができます。（プログラムの秘密化）

●パスワードは、8文字までの英文字、カナ文字、数字、記号を使用することができます。

●パスワードが宣言されていない状態のとき、PASS命令を実行すれば、そのとき計算機内にあるBASICプログラムに対してパスワードが設定され、秘密プログラムになります。

秘密化されたプログラムに対しては、LIST命令、↑や↓キーなど、プログラムの呼び出しにかかわる命令や機能、行の追加・削除などの機能は働きません。

秘密プログラムはNEW、DELETE命令でも消去されず、保護されます。
RENUM命令は無視されます。
また、ファイルエリアへ登録したりテープレコーダーや他の出力機器に出力することもできません。

●秘密プログラムを解除する場合はもう一度同じパスワードを宣言します。(パスワードが違っているときはエラー92になります。)

(注) 8文字以上のパスワードを宣言したときは、頭から8文字のみが有効となり、設定または解除が行われます。

●計算機内にBASICプログラムが入っていないときにPASS命令を実行するとエラー14になり、パスワードは設定されません。

●秘密プログラムでないプログラムを秘密プログラムとしてテープに記録することができます。(CSAVE命令参照)

●" "(Null)はパスワードとして設定できません。

関数

48 PEEK………ピーク

プログラム
マニュアル

省略………PE.

機能 メモリ上の機械語プログラムやデータを直接読み出します。

書式 PEEK 番地

参照 POKE、CALL

説明 ●指定した番地からデータを読み出します。

●番地はØ~65535(&HØ~&HFFFF)の値で指定します。

●読み出されるデータは、Ø~255(&HØ~&HFF)の値になります。

●パスワードが設定されているとき、本命令をマニュアルで実行するとエラー93になります。

〈例〉 4001番地(16進表記)のデータを読み出し、変数Aに入れます。

```
A=PEEK &H4ØØ1
```

基本命令

49 POKE………ポーク

プログラム
マニュアル

省略形……POK.

機能 機械語プログラムやデータをメモリに直接書き込みます。

書式 POKE 番地, データ1, データ2, ……

参照 CALL、PEEK

説明 指定した番地をデータ記憶の開始番地として、データ1、データ2……と、順にメモリに記憶していきます。

●番地はØ~65535(&HØ~&HFFFF)の値で指定します。

●データは1バイト単位で指定していきます。したがって、各データの取りうる値の範囲はØ~255（&HØØ~&HFF）です。

●パスワードが設定されているとき、本命令をマニュアルで実行するとエラー93になります。

〈例〉 &HØ1、&HØ2、&HØ3を7000~7002番地（16進表記）に書き込みます。

```
POKE&H7ØØØ, &HØ1, &HØ2, &HØ3
```

（注） この命令を誤って使用するとBASICプログラムや本機のシステムエリアを破壊し、異常が発生することがあります。

基本命令

50 PRINT……プリント

プログラム　マニュアル

省略形……P.

機能 指定した内容を表示部に表示します。

書式 PRINT [{ 式 / 文字列 } [{ , / ; } { 式 / 文字列 } ………] [;]]

参照 LOCATE、LPRINT、USING、WAIT

説明 ●項目を1つだけ指定したときは、式の値は表示部の右側にツメて表示し、文字は表示部の左端から表示します。

〈例〉
```
10 PRINT "ABCD"
20 PRINT 123
```
RUN ↵

```
RUN
ABCD
                 123.
>
```

ただし、LOCATE命令により、表示開始位置が指定されているときは、その位置から表示します。

●コンマ（,）で区切って2つ以上の項目を指定したときは、表示部を12桁ずつに区切り、最初に指定されている内容から順番に表示していきます。この場合も、12桁の範囲内で式の値は右側にツメて表示し、文字は左側から表示します。なお、数値または文字が12桁を超える場合は次のように処理されます。

①数値が12桁を超える場合（指数表示において、仮数部が7桁以上になる場合）は、仮数部の下位桁が切り捨てられます。

②文字が12桁を超える場合は、先頭から12桁のみを表示します。

〈例〉
```
10 CLS
20 A=123:B=5/9:C$="ABCD"
30 PRINT "A=", A
40 PRINT A, C$, B
```
RUN ↵

```
A=                  123.
         123. ABCD
 5.55555E-01
```

●区切りにセミコロン(；)を使用している場合は、指定された内容を続けて表示します。

〈例〉
```
10 CLS
20 A=123:B=5/9:C$="ABCD"
30 PRINT "A=";A
40 PRINT A;C$;B;"VWXYZ"
```

RUN ↵

```
A=123.
123.ABCD5.555555556E-01V
WXYZ
```

●末尾がセミコロン（；）の場合は、その前に指定されている内容を左につめて表示し、その表示した内容の最後に続く桁が、次のPRINT命令に対する表示開始位置となります。

〈例〉
```
10 A=123:B=45
20 CLS: PRINT "123*45=";
30 C=A*B
40 PRINT C
```

RUN ↵

```
123*45=5535.
```

●PRINTのみで、表示する内容が指定されていないときは、改行を行います。

●LOCATE命令や、末尾がセミコロンのPRINT文で表示開始位置が指定されている場合は、その位置から表示を開始します。

なお、このとき表示する内容の項目が（，）で区切られている場合、最初の項目は12桁の範囲にこだわらずに表示されます。

●1つのPRINT命令で表示に使用する桁数が、255桁を超えるとエラー55になります。

◎PRINT→LPRINT指定

本機はPRINT命令を、必要に応じて印字命令に切り替えて用いることができます。

たとえば、計算機本体のみで使用しているときは、PRINT命令を表示命令として画面に計算結果を表示させ、別売のプリンタを接続しているときは印字命令として計算結果などをプリントさせることができます。

指定・解除

別売のプリンタが接続されているとき、マニュアルあるいはプログラムで

PRINT=LPRINT

の命令を実行すると、PRINT命令はすべてLPRINT命令と同様に働きます。

この機能は

PRINT=PRINT

の命令で解除することができるほか、RUN命令の実行、SHIFT + CA/CLS キーの操作、電源のオフ・オンなどにより解除され、PRINT命令は通常の表示命令にもどります。なお、RUN命令の実行でPRINT=LPRINTの指定は解除されますので、マニュアルでPRINT=LPRINT命令を実行させ、かつ有効に働かせるには、次のような方法を用いてください。

●命令実行後、GOTO命令でプログラムをスタートさせる。(142ページ参照)

●INPUT命令などにより、プログラムがストップしているときに、この命令を実行する。

51 PRINT#…プリント・クロスハッチ

プログラム

省略形……P.#

機能 指定したデータをSIO（シリアル入出力装置）またはカセットテープに出力します。

書式
PRINT #ファイル番号，データ［｛，；｝データ…］［｛，；｝］

参照 OPEN、INPUT#

説明 指定したデータをSIOから出力します。またはカセットテープに記録します。

●OPEN命令で"COM："を指定、または"CAS："のOUTPUTを指定してオープンされているときにのみ有効です。

●ファイル番号は1を指定します。

●"データ"は、一般式（数式、文字式）の他に変数を指定することができます。変数は次のものが使えます。

固定変数	A、X、B$など
単純変数	AB、CD、EF$など
配列の要素	B（1Ø）、C$（5，5）など
配列全体	B（*）、C$（*）など

●"データ"の後ろがコンマ（，）のときは20桁を1ゾーンとして出力します。

データが20桁以内なら1ゾーンの領域の中で、数値は右づめ、文字は左づめで出力します。

文字データが20桁を超えている場合は、必要なゾーン数を確保して出力します。ゾーンの残りの桁はスペースで埋めます。

〈例〉
```
A$="ABC":C=123:D=-5.2E1Ø
PRINT#1,A$,C,D
```

出力
```
ABC␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣123.␣
-5.2E+1Ø␣CRLF
```

●"データ"の後ろがセミコロン（；）のときは、データを続けて出力します。

〈例〉
```
A$="ABC":C=123:D=-5.2E1Ø
PRINT#1,A$;C;D
```

出力
```
ABC␣123.␣-5.2E+1Ø␣CRLF
```

●指定された"データ"をすべて出力すると、最後にCR（&HØD）とLF（&HØA）コードが送られます。ただし、データの最後にコンマ（，）またはセミコロン（；）が指定されているときはCR、LFコードは出力されません。

●数値データの出力フォーマット

①数値データの後ろには1桁分のスペースが付けられます。

②数値の前には符号桁が1桁付きます。数値が負数の場合は、この桁が－（マイナス符号）になり、正数の場合はスペースになります。

③指数形式で出力される場合は、仮数部の後ろに指数部を示す記号(E)、符号、数値2桁（1桁の場合は前にØを付ける）が出力されます。

〈例〉 PRINT#1，12345678987

␣1．234567898E+1Ø␣CRLFを出力

④未定義の変数（CLEAR実行後のABなど）を指定した場合はØが出力されます。

●文字データの出力フォーマット

①指定されているデータをそのままアスキー形式で出力します。

②CHR$（Ø）は" "（Null）になります。

③未定義の変数（CLEAR実行後のAB$など）を指定した場合は" "（Null）が出力されます。

●配列全体を指定したときは、各要素を出力するごとにCR、LFコードを出力します。

たとえば、PRINT#1，B（*）はPRINT#1，B（Ø）：PRINT#1，B（1）：PRINT#1，B（2）……と指定した場合と同じ状態になります。

●配列全体を指定した場合の出力順序は次の例のようになります。

〈例〉 一次元配列 B（Ø）→B（1）→B（2）→……

二次元配列 C（Ø，Ø）→C（Ø，1）→C（Ø，2）→……

C（1，Ø）→C（1，1）→C（1，2）→……

⋮

●配列全体以外の文字変数または文字を指定するとき、コンマ（，）やセミコロン（；）でつなぎますとデータの区切りコードが付きませんので、テープから読み込むときなどに、データを分けることができません。したがって、次のように1命令に1データを指定するようにしてください。

```
PRINT#1，"ABC"
PRINT#1，AB$
```

基本命令

52 RADIAN…ラディアン

プログラム マニュアル

省略形……RAD.

機能 角度単位を“RADIAN”（ラディアン）に設定します。

書式 RADIAN

参照 DEGREE，GRAD

説明 三角関数、逆三角関数，座標変換関数で扱う角度の単位を“RADIAN”単位〔rad〕に設定します。

（1直角＝π/2〔rad〕）

基本命令

53 RANDOMIZE…ランダマイズ

プログラム
マニュアル

省略形……RA.

機能 RND命令の使用に先立って乱数のタネを植えつけるものです。

書式 RANDOMIZE

参照 RND

説明 RND命令により乱数を発生させる場合、電源を入れ直しますと常に同じ乱数系列を発生します。しかし、電源を入れた後にRANDOMIZE命令を実行すれば、同じ乱数を発生することは、まず、なくなります。

基本命令

54 READ………リード

プログラム

省略形……REA.

機能 DATA文に続いて指定されているデータを変数に読み込みます。

書式 READ 変数[,変数]………

参照 DATA、RESTORE

説明 ●変数にデータを代入する方法の1つであり、変数はREAD文で指定し、データはDATA文で指定します。

〈例〉

```
 50 READ A, B
 110 DATA 3, 4, 5
 120 DATA 60, "G=", "␣H="
 200 READ C, D, E$, F$
```

Aには3、Bには4が代入されます。

Cには5、Dには60、E$には"G="、F$には"␣H="が代入されます。

●1つのプログラムの中に何回でもREAD文およびDATA文を書くことができますが、データは何行に分けて書いても一連のデータと見なされ、若い番号の行のデータから順番に変数に代入されます。また、複数のプログラムが書き込まれ、それぞれのプログラムにDATA文がある場合でも、それは一連のデータと見なされます。

したがって、それぞれのプログラム内のデータを使用するときは、プログラムの最初の行に、RESTORE命令を書き込んでください。

●データと変数はその型（数値変数か文字変数かの型）が一致していなければなりません。

●読み込もうとして、読み込むデータがない場合はエラー53になります。

●DATA文では文字データを"　"で囲まずに指定できますが、この場合、データの前後にスペースが指定されていても、それはないものと見なします。

基本命令

55 REM…………リマーク

プログラム

省略形……なし

機能 プログラムに注釈を入れます。

書式 REM 注釈

参照 この命令はプログラムの実行には関係なく、プログラムリストなどをわかりやすくする目的でプログラムの先頭や途中に注釈を入れておくためのものです。

●同じラインで、REMの後に書かれている内容は、すべて注釈とみなされプログラム実行時は無視されます。

●REMの代わりに'（シングルクォーテーション）を使用することもできます。

(注) コロン（：）の後に'を入れる場合は、間にスペースを入れてください。

〈例〉

```
  1Ø  REM  キンリ  ケイサン
       ：
 2ØØ  'サブルーチン
       ：
```

基本命令

56 RENUM……リナンバー

マニュアル

省略形……REN.

機能 プログラムの行番号を付け直します。（PROモードのマニュアル操作でのみ有効）

書式 RENUM［新行番号］［，［開始行］［，増分］］ ↵

説明 ●"開始行"で指定したプログラムの行番号を"新行番号"に書き換え、それ以降の行番号を、"増分"で指定した値にしたがって順次書き換えていきます。

●"開始行"で指定した行番号がプログラム内に存在しない場合は、エラー40になります。

●新行番号、開始行、増分を省略した場合、開始行はプログラムの最初の行、新行番号、増分はそれぞれ10が指定されたものとみなされます。

〈例〉 RENUM ↵

プログラム全体の行番号を10から10ステップきざみで付け直します。

RENUM 1ØØ，5Ø，1Ø ↵

行番号5Øが1ØØになり、それ以後10ステップきざみで最後まで付け直します。

●RENUM命令は、GOTO、GOSUB、IF～THEN、ON～GOTO、ON～GOSUB、RESTORE命令などで参照している行番号も新しい行番号に対応させて自動的に付け直します。なお、参照する行番号がプログラム内に存在しないときはエラー40になります。

●上記の命令で参照している行番号またはラベルの中に行番号やラベルとみなせない文字や記号が入っていると、それより後に書かれている行番号の付け直しは行いません。

〈例〉 GOTO 1Ø*2

ON A GOTO"ABC"，2ØØ，AB，4ØØ

～で示す部分は行番号やラベルとなりません。＿で示す部分は行番号とみなされ、付け直しを行います。

●新しく付け直す行番号が65279を超える場合はエラー41になります。

●実行の順番が変わるような指定（たとえば10、20、30の3つの行があるとき、RENUM15，30を実行する場合）を行うとエラー43になります。

●長いプログラムに対してRENUM命令を実行すると、少し時間がかかります。表示部右端に＊を1つ表示しているときは BREAK ON キーで中断すると実行前のプログラムの状態にもどりますが、＊を2つ表示しているときは BREAK ON キーは無視されます。

基本命令

57 RESTORE…リストア

プログラム

省略形……RES.

機能　READ命令に続く変数に読み込まれるデータの順番を変えます。

書式　RESTORE［読み込み開始行］

参照　READ、DATA

説明　READ命令の実行時、DATA命令で指定されているデータのどのデータを読み込むかは常に計算機に記憶されていますが、この読み込むデータの順番を強制的に変えるときに使用します。

● “読み込み開始行”を指定すると、その行のDATA命令または指定した開始行以降の最も小さい行番号のDATA命令から読み込みを開始します。

● “読み込み開始行”を省略するとプログラムの最初のDATA命令のデータから読み込みを開始します。

● “読み込み開始行”は、行番号またはラベルで指定します。

関数

58 RIGHT$…ライトドル

プログラム　マニュアル

省略形……RI.

機能　文字列の右側から何文字かを取り出します。

書式　RIGHT$（文字列，式）

参照　LEFT$、MID$

説明　指定された文字列の右から、式の値で指定された桁数（文字数）だけ、文字を取り出します。

●式の値はØ～255の範囲の整数でなければなりません。

〈例〉
```
1Ø  A$="ABCDE"
2Ø  B$=RIGHT$(A$, 3)     ←A$の文字列の右側3文字すなわちCDEを取り出しB$に代入します。
3Ø  PRINT  B$
```

59 RND…………ランダム

省略形……RN.

プログラム
マニュアル

機能 乱数（擬似乱数）を発生させます。

書式 RND式

参照 RANDOMIZE

説明 RND命令は乱数を発生させる機能を持つ関数です。

RND Xにおいて Xの値により次のような乱数を得ることができます。

●Xが2以上の場合

Xが整数のとき：1からXの値以下の乱数を発生します。

（1≦RND X≦X）

Xが小数部を含むとき：1からXの整数部に1を加えた値以下の乱数を発生します。

（1≦RND X≦（INT X）+1）

ただし、この場合は小数部の値によって乱数の発生に差が生じます。

●Xが負数の場合：同じ乱数（乱数列）を発生させるために、初期値を一定にします。

●Xが0から2未満の場合：1未満の乱数を発生します。

（0<RND X<1）

●乱数の有効桁数は10桁です。

〈例〉

```
 10 CLS
 20 USING "##"
 30 FOR A=0 TO 2
 40 Y=RND -1
 50 FOR B=0 TO 3
 60 C= RND 9
 70 PRINT C;
 80 NEXT B
 90 PRINT: NEXT A
100 END
```

40行は同じパターンの乱数を発生させるために入れています。
これにより、3回同じ乱数列が表示されます。
40行を削除してプログラムを実行して見てください。
同じ乱数が発生されなくなります。
なお、変数Yは、ほかに使用されていない変数ならばY以外でもかまいません。

プログラムを実行すれば同じ乱数列が表示されます。

```
7 1 4 3
7 1 4 3
7 1 4 3
```

3回表示されます。

60 RUN…………ラン

マニュアル

省略形……R.

機能 プログラムの実行を開始します。（RUNモードのマニュアル操作でのみ有効）

書式 （1）RUN ↵

（2）RUN {行番号 / ラベル} ↵

参照 GOTO

説明
- 書式（1）の形では、プログラムの一番小さい行番号から実行が開始されます。
- 書式（2）の形では、指定した行あるいはラベルが書かれている行からプログラムの実行が開始されます。（ラベルについては143ページ参照）
- 指定した行番号あるいはラベルがない場合はエラー40になります。
- プログラムに同じラベルが2個以上書かれているときは、行番号の小さいほうが実行されます。
- RUN命令によりプログラムの実行が開始されたときの計算機の状態については142ページをご覧ください。

61 SAVE………セーブ

プログラム
マニュアル

省略形……SA.

機能 BASICプログラムをファイルエリアに登録します。

書式 SAVE "ファイル名"

参照 LOAD、FILES

説明
- BASICプログラムをファイル名をつけて登録します。
- ファイル名は8文字以下の名前と、拡張子を3文字まで指定することができます。
- 拡張子の記述を省略した場合は「.BAS」になります。
- 既存のファイル名を指定した場合は、そのファイルの書き換えになります。ただし、既存のファイル名がTEXT（テキスト）ファイルの場合はエラー96になります。
- 計算機内のプログラムが秘密化されているときや、プログラムがないときは、SAVE命令は無視されます。

基本命令

62 STOP……ストップ

プログラム

省略形……S.

機能 プログラムの実行を一時停止させます。

書式 STOP

参照 CONT

説明 プログラムにバグ（虫…誤りのこと）が発生したとき、プログラムの実行を途中で止めて、変数の状態を調べたり、変数に数値などを代入したりしてバグを探しますが、このようなとき、プログラムを止めたい位置にSTOP命令を書き込みます。

●この命令が実行されますと「BREAK　IN　2ØØ」のように、実行した行番号を伝えるブレークメッセージを表示して停止します。

●この命令により停止したプログラムの実行を再開させるときはCONT命令を使用します。

関数

63 STR$……ストリングドル

プログラム
マニュアル

省略形……STR.

機能 数値を文字列に変換します。

書式 STR$ 式

参照 VAL

説明 数値を文字列の形に変換します。

たとえば、B=1234のとき

A$=STR$ B

として実行すれば、A$には"1234"という文字列が代入されます。

基本命令

64 TROFF……トレースオフ

プログラム
マニュアル

省略形……TROF.

機能 トレースモードを解除します。

書式 TROFF

参照 TRON

説明 TRON命令で設定されるトレースモードを解除します。（TRON参照）

基本命令

65 TRON……トレースオン

プログラム マニュアル

省略形……TR.

機能 トレースモードを設定します。

書式 TRON

参照 TROFF

説明 トレースモードでは、プログラムが1行実行されると、その実行した行番号を表示して停止します。次の行を実行するときは ↓ キーを押します。↓ キーでプログラムを1行ずつ実行させていくことができますので、プログラムがどのように実行されていくかをたどることができます。(135ページ参照)これにより、プログラムが正しく実行されるかどうかチェックすることができます。

●トレースモードはTROFF命令の実行、SHIFT + CA キーの操作、電源オフ・オンなどで解除されます。

基本命令

66 USING……ユージング

プログラム マニュアル

省略形……U.

機能 数値や文字などを表示あるいは印字するときのフォーマットを指定します。

書式 (1) USING"フォーマット"
(2) USING

参照 PRINT、LPRINT

説明 PRINT命令による表示の形、LPRINT命令による印字の形を指定する命令です。また、マニュアル計算の結果が数値の場合には、このUSINGのフォーマットに従います。

●表示フォーマットは、書式(1)の形でUSING命令に続く"　"内に、次の記号を用いて指定します。また、指定の解除は書式(2)で行います。

#	数値の桁数を指定 整数部桁数：2～11桁(符号を含む) 指定桁より数値のほうが少ないときはその分だけスペース表示になり、多いときはエラーになります。12桁以上指定した場合はすべて11桁の指定とみなされます。 小数部桁数：Ø～12桁(指数方式による表示のときはØ～9桁) 指定桁より数値のほうが少ないときはその分だけØ表示になり、多いときはその分だけ切り捨てられます。
.	小数点の表示を指定(整数部と小数部の区切りを指定)
∧	数値の指数方式による表示を指定 整数部の指定桁にかかわりなく、仮数部の整数桁は2桁(符号を含む)になります。 小数部の桁数を9桁以上に指定した場合、仮数部の小数桁は9桁になります。
&	文字列の桁数を指定 指定桁より文字数が少ない場合はその分スペース表示になり、文字数が多い場合は指定された桁分だけ表示します。

〈例〉 ① USING "###"　符号を含めて整数3桁を表示

② USING "###."　符号を含めて整数3桁と小数点を表示

③ USING "###.##"　符号を含めて整数3桁と小数点と小数点以下2桁を表示

④ USING "##.##^"　小数点以下2桁までの指数方式での表示

〔このとき、仮数部の整数は符号を含めて2桁、指数部は符号を含めて4桁（E　OO）が自動的に確保されます。〕

⑤ USING "&&&&&&"　文字を6桁表示

⑥ USING "###&&&&"　数値と文字を同時に指定

⑦ USING　フォーマット指定を解除

●USING命令はPRINT文の中でも使用することができます。

〈例〉

```
10 B=-10.8:C=10.7703
20 PRINT USING "&&&###";"B=";B,
   "C=";USING "###.###";C
```

●USING命令で表示フォーマットの指定を行いますと、以降に実行されるPRINT命令、LPRINT命令には、すべてそのフォーマット指定が有効になります。
したがってフォーマット指定が不要なときは書式（2）の形で指定を解除してください。
なお、フォーマット指定はRUN命令の実行や、SHIFT + CA キーの操作などでも解除されます。

関数

67 VAL…………バリュー

プログラム　マニュアル

省略形……V.

機能 文字列を数値に変換します。

書式 VAL　文字列

参照 STR$

説明 数値を表すことのできる数字（0〜9）、符号（+、−）、指数部を示す記号（E）で構成されている文字列および先頭に&Hがついた16進数を表わすことのできる文字列を数値（10進数）に変換する関数です。
一つの文字列の中に数値とすることのできない文字や記号が含まれている場合、それから右の文字列は無視されます。

〈例〉 A=VAL"−120"　Aに−120が代入されます。

B=VAL"3.2*4="　Bに3.2が代入されます。

C=VAL"&HFF"　Cに255が代入されます。

68 WAIT………ウエイト

プログラム
マニュアル

省略形……W.

機能 PRINT命令によるプログラムの停止時間を指定します。

書式 （1）WAIT 式

（2）WAIT

参照 PRINT

説明 ●書式（1）の形において、式で時間を指定すると、PRINT命令を実行するたびに指定時間だけプログラムが停止し、その時間が経過するとプログラムは自動的に再開します。

●式の値はØ～65535の範囲で指定できます。

なお、式の値1は約1／64秒に相当します。

●時間を無限にしたいときは、書式（2）の形で時間指定を解除してください。この場合、プログラムを再開するには ↵ キーを押す必要があります。

●本機の電源を入れた時点、またはRUN命令を実行したときはWAIT Ø（停止時間Ø）に設定されます。

注）一般のパソコンではWAIT指定はできません。パソコンでは通常次のようにFOR～NEXTを用いて、時間の調整を行います。

〈例〉 5Ø FOR J=1 TO 5ØØ:NEXT J

本機には、簡単な外付インターフェイスにより入力および出力の可能な制御ポートと、XYプロッタやアームロボットを制御できるミニI／O機能があり、それぞれポート制御命令とデータ出力命令によって実現できます。

これらの命令は、本機に外付インターフェイスを接続し、上記に示すような機器を制御するとき以外には使用しないでください。

ミニI／Oに関する命令

69 INP…………イン・ポート

省略形……なし

プログラム　マニュアル

機能　入力ポート（Xin，Din，ACK）からの入力関数です。

書式　INP

参照　OUT

説明　ポート制御命令で、INP命令は現在の入力ポートの状態を読み取って、Ø～7の値で返します。入力ポートはXin，Din，ACKの3つからなり，それぞれXin=4、Din=2、ACK=1の重みを持ちます。すなわち各入力ポートのそれぞれの状態を2進数の各桁と見なしたときの値を10進数に変換してØ～7の値を返します。なお、信号レベルはHi=1、Lo=Øとします。

〈例1〉 Xin=Lo，Din=Hi，ACK=Hi のとき
Ø×4+1×2+1×1=3 で
INPの値は3となります。

〈例2〉 5Ø IF INP AND 2 THEN5Ø
Dinの値がLoレベルであれば次の行に実行が移り、Hiレベルであれば5Ø行をくり返し実行します。

ミニI／Oに関する命令

70 OUT…………アウト・ポート

省略形……OU.

プログラム　マニュアル

機能　出力ポート（Busy，Dout，Xout）への出力命令です。

書式　OUT 式

参照　INP

説明　ポート制御命令で、OUT命令は10進数（Ø～7）で指定した値を出力ポートBusy、Dout、Xoutに出力します。指定する値は、出力ポートをそれぞれBusy=4、Dout=2、Xout=1 の重みで加算した値です。すなわち各出力ポートのそれぞれの状態を2進数の各桁と見なしたときの値を10進数で指定します。なお、信号レベルはHi=1、Lo=Øとします。

〈例〉 OUT6
6=1×4+1×2+Ø×1 で
Busy=Hi，Dout=Hi，Xout=Lo となります。

（注） OUT命令で出力した内容は次のOUT命令が実行されるまで、保持されます。ただし、次の操作をおこなった場合は出力ポートの状態が変化します。

⑴ プリンタ、カセットなどに関する命令を実行したとき

LPRINT	PRINT#
LLIST	INPUT#
CSAVE	SHIFT ＋ P↔NP
CLOAD	

⑵ その他、次の操作を行ったとき

- 電源を入れた直後
- RUN実行直後
- マニュアル計算実行

ただし、あらかじめOPEN"LPRT:" ↵ と操作しておけば出力ポートの状態は保持されます。

ミニI／Oに関する命令

71 OPEN………オープン

省略形……OP.

プログラム　マニュアル

機能　出力デバイスを指定します。

書式　OPEN"LPRT:"

参照　CLOSE、LPRINT、LLIST

説明　OPEN"LPRT:"実行後のLPRINT、LLIST命令は、パラレルポートに対しての出力命令になります。

ミニI／Oに関する命令

72 CLOSE……クローズ

省略形……CLOS.

プログラム　マニュアル

機能　デバイス指定を解除します。

書式　CLOSE

参照　OPEN、LPRINT、LLIST

説明　CLOSE命令によってOPEN状態を解除し、それ以後のLPRINT、LLIST命令はCE−126P（専用プリンタ）に対しての命令になります。

ミニI／Oに関する命令

73 LLIST……ラインリスト

プログラム　マニュアル

省略形……LL.

機能　プログラム内容をパラレルポートより送出します。

書式　（1）LLIST

（2）LLIST {行番号 / ラベル}

（3）LLIST開始行－終了行

参照　OPEN、CLOSE

説明　データ出力命令で、OPEN"LPRT："により、パラレルポートが指定されているときに、プログラムをアスキーコードで送出します。

CLOSEされているときは、プリンタ（CE－126P）でプログラムを印字する命令になります。（223ページ参照）

ミニI／O命令

74 LPRINT…ラインプリント

プログラム　マニュアル

省略形……LP.

機能　指定した内容をパラレルポートから送出します。

書式　LPRINT [{式 / 文字列} [{, / ;} {式 / 文字列}……] [;]]

参照　OPEN、CLOSE

説明　OPEN"LPRT："によりパラレルポートが指定されているときに、指定した内容をパラレルポートからアスキーコードで送出します。

CLOSE（クローズ）されているときは、プリンタ（CE－126P）の印字命令になります。

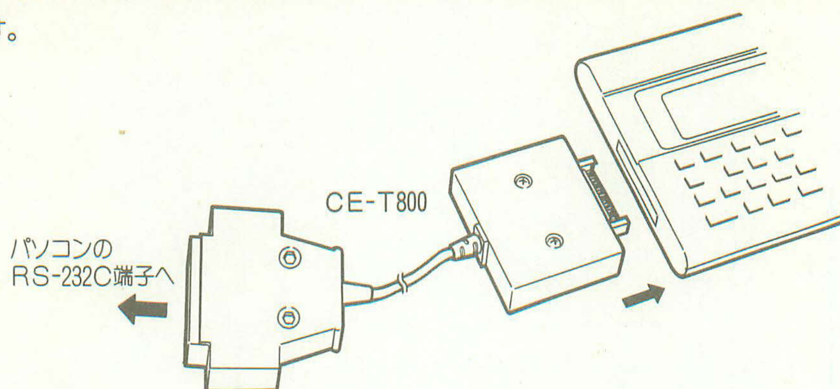
第9章　その他

1.通信用ケーブルCE-T800のご案内

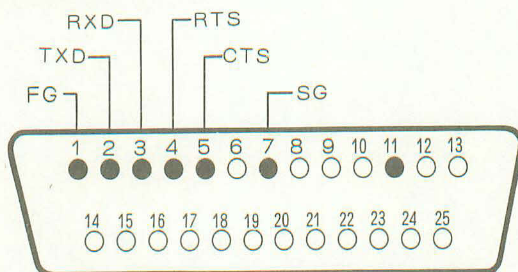
CE-T800は、本機とパソコンなどとの通信(シリアル方式)を行うためのケーブルです。
このケーブルを使用することにより、本機とパソコンとの間でTEXTモードのSIO機能を使用したプログラムの入出力、機械語モニタを使っての機械語の入出力などを行うことができます。

このケーブルは、内部に本機とパソコンのRS-232C端子の信号レベルを合わせるための回路が入っています。
シリアル方式によるパソコンとの通信には、パソコンのRS-232C端子を使用しますが、本機とRS-232Cでは信号の電圧レベルが異なり、RS-232Cの信号を直接本機に加えると本機が壊れます。
したがって、この信号レベルを合わせるための回路がCE-T800に入っています。

注) CE-T800の端子に指などで触れないでください。静電気などにより内部回路が壊れることがあります。



【CE-T800のコネクタ信号〈DB-25(W)〉】



信号名

RS-232C信号			本機から見た信号の方向	機能概要
ピン番号	信号名	記号		
1	フレーム接地	FG	—	
2	送信データ	TXD(SD)	入力	本機に送られてくるデータ信号
3	受信データ	RXD(RD)	出力 注2)	本機から送信するデータ信号
4	送信要求	RTS(RS)	入力	データを送信するとき、この信号がハイレベルならば送信を行い、ローベルになったときは送信を停止します。
5	送信可	CTS(CS)	出力 注2)	本機が受信可能なときはハイレベルになり、受信できないときはローレベルになります。
7	信号用接地	SG	—	入出力装置間の基準電位を合わせます。
11	—	NC	—	本機では使用していません。

注1）CE-T800は、パソコンのRS-232C端子につないで使用できるよう、パソコンからの出力信号(2番、4番)は、本機の入力信号に、またパソコンの入力信号(3番、5番)へは、本機の出力信号が接続されています。

注2）これらの出力信号は、下記以外の状態では不定です。
①BASICモードでSIOがオープンしているとき。
②機械語モニタモードでR命令およびW命令を実行するとき。
③TEXTモードでSIOの送受信をするとき。

【パソコンとの接続例】
本機とMZ-2861との間でプログラム転送をする場合
パソコン側はMS-DOS※を使用し、TEXTプログラム（アスキー形式）および機械語プログラム(インテル・ヘキサ方式)のファイルを転送するものとします。

※MS-DOSは米国マイクロソフト社の登録商標です。

1．本機からMZ-2861への転送
①まず、双方の転送フォーマット(通信条件)の設定を行います。
本機(TEXTモードのSIOのFormat機能を用いてください。)
通信速度……………1200 bps
ワード長……………8ビット
ストップビット数……1ビット
パリティ……………なし(none)
区切りコード…………CR　LF
テキスト終了コード…1A
行番号…………………YES
MZ-2861
>SETUP ↵
(RS／CSは有効にしてください。)
②MZ-2861を読み込み可能状態にします。
>COPY　AUX　〈ファイル名〉／A ↵
③本機から、プログラムを送信してください。
TEXTプログラム……TEXTモードのSIOのSave機能を使用
機械語プログラム………機械語モニタのW命令を使用
通信が完了すると、プログラムをディスクに登録して終了します。

2．MZ-2861から本機への転送
①双方の転送フォーマット(通信条件)の設定を行ってください。
②本機を読み込み可能状態にします。
TEXTプログラム……TEXTモードのSIOのLoad機能を使用
機械語プログラム………機械語モニタのR命令を使用
③MZ-2861から、プログラムを送信してください。
>COPY／B　〈ファイル名〉　AUX ↵

2. 電池の交換について

本機は、本体動作用とメモリ保護用の２種類の電池を使用します。それぞれ次の方法で交換してください。

(1)電池を交換する前に

別売のプリンタ／カセットインターフェイス(CE-126P)とテープレコーダ(CE-152)をお持ちの場合は、電池交換の前にプログラムやデータをテープに記録しておくことをお勧めします。

本機はBASICプログラムの他に、機械語プログラム、TEXTプログラムをそれぞれ独立して記憶できるようになっています。なお、BASICプログラムとTEXTプログラムは、常にファイルエリアに登録しておくような使いかたをしておけば、この2つについてはファイルエリアを一括して記録することによって、手間をはぶくことができます。これらは、次の命令または機能でテープに記録することができます。

BASICプログラム	CSAVE命令(読み込みはCLOAD)
データ（変数の内容）	PRINT#命令(読み込みはINPUT#)
TEXTプログラム	TEXTモードのCMTのSave機能(読み込みはLoad機能)
機械語プログラム	CSAVE　M命令(読み込みはCLOAD　M)
ファイルエリアのプログラムファイル	TEXTモードのCMTのSave機能でファイル名の指定を＊．＊として記録(ファイルエリアの内容が一括して記録されます。)(読み込みはLoad機能でファイル名を＊．＊に)

(2)動作用電池の交換

● 動作用電池の寿命について

本体動作用の電池が消耗して、規定電圧以下になると画面右上に BATT シンボルが点灯します。

このシンボルが点灯したときは、一度 OFF キーで電源を切り、ON キーで電源を入れてください。それでもなおこのシンボルが消えないときは、速やかに次の手順で新しい電池と交換してください。このシンボルが点灯した状態で使用し、さらに電池が消耗しますと、本機の電源が切れて何も動作しなくなります。

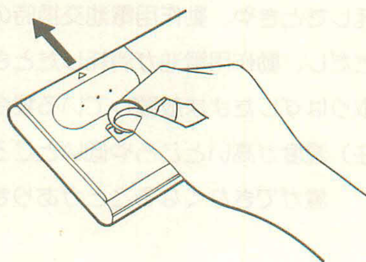
注）機械語実行中は、電池が消耗しても BATT シンボルが点灯しませんので、実行させたまま長期間放置しないでください。機械語を実行したまま長期間放置しますと、電圧が低下し、正常な動作をしなくなる恐れがあります。

なお、機械語実行中に誤動作等が発生した場合は電池の消耗が考えられますので、リセットスイッチ❺で実行を止め、BATT シンボルの点灯または電源が切れる場合は速やかに電池を交換してください。

●動作用電池の交換のしかた

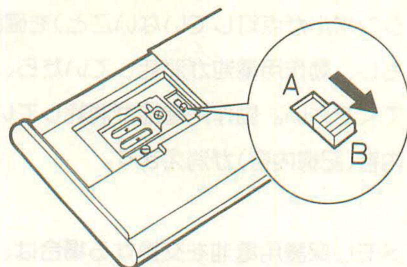
動作用電池を交換する場合は、次の手順に従ってください。

①OFFキーを押して電源を切ってください。



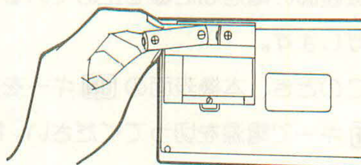
②本機裏面の電池ぶた⑬を止めているネジをコインやネジ回しなどでゆるめ、電池ぶた⑬を取りはずします。

注）このとき、本機表面のONキーを押さないように注意してください。誤って押した場合はOFFキーで電源を切ってください。電源が入った状態で③の操作を行うと、メモリの内容が消えたり壊れたりします。

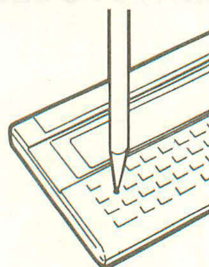


③メモリ保護スイッチ⑮をB側に寄せてください。

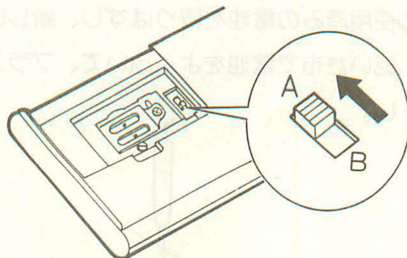
注）このスイッチをB側に寄せないで電池交換を行うと、メモリの内容が消えたり、壊れたりします。



④乾電池を新しいものと交換します。(乾電池は4本とも交換してください。)
乾電池の⊕・⊖をまちがえないように、⊖(マイナス)側から入れます。



⑤電池ぶた⑬を取り付け、本機表面のリセットスイッチ⑤をボールペンなどで押し、何も表示されないことを確認してください。
もし、何か表示されたときは、もう一度リセットスイッチ⑤を押してください。



⑥電池ぶた⑬をはずし、メモリ保護スイッチ⑮をA側に寄せてください。

⑦電池ぶた⑬を取り付け、ゆるまないようにネジをしめます。

注）メモリ保護スイッチ⑮をA側にしないで本機の電源を入れても、本機は何も動作しません。このときは、もう一度電池ぶた⑬をはずしてスイッチをA側にしてください。

(3)メモリ保護用電池の交換

●メモリ保護用電池の寿命について

本機のメモリ保護用電池の寿命は、常温(約20℃)において約5年です。この間、動作用電池が消耗したときや、動作用電池交換時のメモリ内容(記憶内容)の保護ができます。

ただし、動作用電池が消耗したとき、すぐに交換せずに放置している場合、または動作用電池を取りはずしたまま放置している場合では、メモリ保護用電池は速く消耗します。

注) 温度が高いところや低いところなど、使用環境によっては電池寿命が短くなり、メモリの保護ができなくなることがあります。

●メモリ保護用電池の交換のしかた

メモリ保護用電池を交換する場合は、動作用電池が消耗していないこと(電源を入れて、BATTシンボルが点灯していないこと)を確認してください。

もし、動作用電池が消耗していたら、先に動作用電池を交換してからメモリ保護用電池を交換してください。動作用電池が消耗している状態でメモリ保護用電池を取りはずしますと、メモリの内容(記憶内容)が消えます。

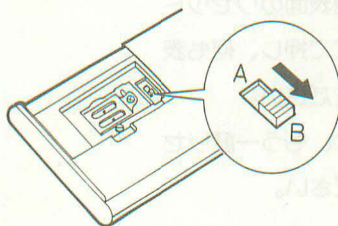
メモリ保護用電池を交換する場合は、次の手順に従ってください。

① OFF キーを押して電源を切ってください。

②本機裏面の電池ぶた⑬を止めているネジをコインやネジ回しなどでゆるめ、電池ぶた⑬を取りはずします。

(このとき、本機表面の ON キーを押さないように注意してください。誤って押した場合は、OFF キーで電源を切ってください。電源が入った状態で次の③の操作を行うと、メモリの内容が消えたり壊れたりします。)

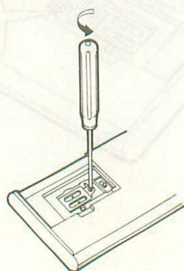
③メモリ保護スイッチ⑮をB側に寄せてください。

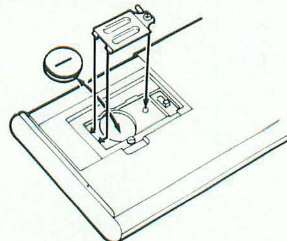


④小型のネジ回しで電池押さえのネジをはずし、電池押さえをはずしてください。

⑥使用済みの電池を取りはずし、新しいリチウム電池を1個入れます。

乾いた布で電池をよくふいて、プラス・マイナス(⊕・⊖)をまちがえないように入れてください。





注) 動作用電池がはずれないように注意してください。

⑥電池押さえを取り付け、ネジをしめてください。

⑦メモリ保護スイッチ⑮をA側に寄せてください。

⑧電池ぶた⑬を取り付け、ゆるまないようにネジをしめます。

(4)電池交換の際のご注意

電池交換は前記の手順で行っていただくか、あるいはお買い上げの販売店もしくは別紙一覧表に記載のサービス窓口にお申し付けください。有料にて交換いたします。

● 本機には次の新しい電池をご使用ください。

本体動作用　：単3タイプ乾電池　　4本

メモリ保護用：リチウム電池CR2032　1個

〈電池使用上のお願い〉

電池は誤った使いかたをしますと、液もれや、破裂することがありますので、次の点について特に注意してください。

1. 電池のプラス⊕とマイナス⊖の向きを本機の表示通り正しく入れてください。
2. 新しい電池と一度使用した電池を混ぜて使用しないでください。
3. 種類のちがう電池を混ぜて使用しないでください。同じ形状でも電圧の異なるものがあります。
4. 電池は使えなくなったり、長時間使わないときは本機から取りはずしてください。
5. 電池には充電式と充電式でないものがあります。電池の注意表示をよく見てご使用ください。

注）● 電池は幼児の手がとどかないところに置いてください。万一飲み込んだ場合にはただちに医師にご相談ください。

● 消耗した電池をそのままにしておきますと、電池の液がもれて計算機をいためることがあります。

● 電池を火中に投入しないでください。火中に投入すると破裂する恐れがあります。

3. 主なキーの主な機能

次に主なキーの主な機能を説明します。

キー	機能
BREAK ON	●電源を入れるときに押します。 ●プログラム実行中、プログラムを一時停止状態(BREAK:ブレーク)にします。 ●CSAVEやLOADなどの命令実行中は、その実行を中止します。 ●統計計算では、機能の選択画面に戻すときにも使用します。 ●TEXTモード、CASLモードでは、メニュー画面、メインメニュー画面に戻すときにも使用します。
OFF	●電源を切ります。
BASIC	●RUNモードとPROモードを切り替えます。 ●他のモードからRUNモードにします。
TEXT	●TEXTモードにします。 ●TEXTのメインメニュー画面にします。
SHIFT + CASL	●CASLモードにします。 ●CASLのメニュー画面にします。
SHIFT	●各キーの薄茶色で書かれている機能を使うとき、このキーを押さえたまま、それぞれのキーを押します。(2nd F キー参照)
CAPS	●アルファベットの大文字と小文字の入力モードを切り替えます。(画面左上の"CAPS"シンボルの点灯、消灯を行います。)
小文字	●カナの入力モードのとき、小さいカナ文字(ァィゥェォャュョッ)を入力したいときに押します。("小"シンボルの点灯、消灯を行います。)
カナ	●カナ入力モードの設定、解除を行います。("カナ"シンボルの点灯、消灯を行います。)
TAB	●カーソルを決められた桁数だけ移動させます。 RUN、PROモードでは7桁ずつ移動します。 TEXTモードのエディタでは、最初8桁、2回目は6桁、3回目以降は7桁ずつ移動させます。
►	●カーソルを右に移動させます。 ●プレイバックを行います。 ●プログラムが表示されているとき、カーソルが表示されていない場合にカーソルを呼び出します。 ●マニュアル計算などでのエラーを解除します。
◄	●カーソルを左に移動させます。 ●その他、 ► と同じ。
ANS	●ラストアンサーを呼び出します。
CONST	●定数計算の定数と計算命令を設定します。("CONST"シンボル点灯) 2nd F CONST (SHIFT + CONST)と押すと、設定されている定数を表示します。
INS	●現在のカーソル位置に、1文字分のスペースを挿入します。
SHIFT + DEL	●現在のカーソル位置の文字を削除します。

キー	機　能
[BS]	●カーソル位置の１字前の文字を削除します。
[2nd F]	●各キーの薄茶色で書かれている機能を使うとき、各キーを押す前にこのキーを押します。（[SHIFT] キー参照）
[CLS]	●入力中の内容や、表示をクリア(消去)するときに押します。 ●エラーを解除します。
[SHIFT] ＋ [CA]	●計算機の状態を解除します。（クリアオール） ・プログラムの実行が一時停止状態にあるとき、実行を中止させます。 ・表示内容などを消去します。 ・表示フォーマット指定(ＵＳＩＮＧ指定)を解除します。 ・トレースモードを解除(ＴＲＯＦＦ状態に)します。 ・エラーを解除します。 その他
[↵]	●プログラムの行の終了を指定します。 ●プログラムを計算機に書き込みます。 ●マニュアル計算の実行、あるいはＢＡＳＩＣ命令などのマニュアル操作による実行を行います。 ●ＰＲＩＮＴ命令やＩＮＰＵＴ命令で一時停止しているプログラムの再スタートなど、プログラムの再開に使用します。
[SHIFT] ＋ [P↔NP]	●プリンタが接続され、電源が入っているとき、プリントモードの設定、解除を行います。（"ＰＲＩＮＴ"シンボルの点灯、消灯を行います。）

[↓] [↑] キーについては、モードの指定および計算機の状態によって変わります。ＢＡＳＩＣモード(ＲＵＮ、ＰＲＯ)では次のようになります。

モード	状　態	[↓]	[↑]
ＲＵＮ	プログラム実行中	無効	無効
	プログラムの一時停止中 ＷＡＩＴが無限のＰＲＩＮＴ、ＩＮＰＵＴ命令実行中のブレーク状態	次の行を実行 （１行ずつ実行して停止します）	押している間、実行している行あるいは実行した行を表示
	プログラム実行時のエラー状態	無効	押している間、エラーが発生した行を表示
	トレースモードオン状態	トレースを実行	押している間、実行している行あるいは実行した行を表示
ＰＲＯ（ＲＵＮモードからＰＲＯモードに切り替え、プログラムが表示されていないとき）			
	プログラムの一時停止中	停止している行を表示	同　左
	エラー発生後	エラーが発生した行を表示	同　左
	その他	先頭行を表示	最終行を表示
ＰＲＯ（プログラム行が表示されているとき）			
		次のプログラム行を表示	１行前の行を表示

4．計算範囲

加減乗除算

被演算数、演算数、結果

$\pm 1 \times 10^{-99} \sim \pm 9.999999999 \times 10^{99}$および0

関数計算

関数	計算範囲	備考
SIN x COS x TAN x	DEG　: $\lvert x\rvert < 1\times10^{10}$ RAD　: $\lvert x\rvert < \frac{\pi}{180}\times10^{10}$ GRAD : $\lvert x\rvert < \frac{10}{9}\times10^{10}$ ただしtan x において次の場合は除く DEG　: $\lvert x\rvert = 90(2n-1)$ RAD　: $\lvert x\rvert = \frac{\pi}{2}(2n-1)$ GRAD : $\lvert x\rvert = 100(2n-1)$ （n は整数）	
ASN x ACS x	$-1 \leqq x \leqq 1$	$\sin^{-1}x$ $\cos^{-1}x$
ATN x	$\lvert x\rvert < 1\times10^{100}$	$\tan^{-1}x$
HSN x HCS x HTN x	$-227.9559242 \leqq x \leqq 230.2585092$	$\sinh x$ $\cosh x$ $\tanh x$
AHS x	$\lvert x\rvert < 1\times10^{50}$	$\sinh^{-1}x$
AHC x	$1 \leqq x < 1\times10^{50}$	$\cosh^{-1}x$
AHT x	$\lvert x\rvert < 1$	$\tanh^{-1}x$
LN x LOG x	$1\times10^{-99} \leqq x < 1\times10^{100}$	$\ln x = \log_e x$
EXP x	$-1\times10^{100} < x \leqq 230.2585092$	e^x　$e \fallingdotseq 2.718281828$
TEN x	$-1\times10^{100} < x < 100$	10^x
RCP x	$\lvert x\rvert < 1\times100^{100}$　　$x \neq 0$	$\frac{1}{x}$
SQU x CUB x SQR x CUR x	$\lvert x\rvert < 1\times10^{50}$ $\lvert x\rvert < 2.154434690\times10^{33}$ $0 \leqq x < 1\times10^{100}$ $\lvert x\rvert < 1\times10^{100}$	x^2 x^3 $\sqrt{x}$ $\sqrt[3]{x}$
y^x(y^x)	● $y > 0$のとき $-1\times10^{100} < x \log y < 100$ ● $y = 0$のとき $x > 0$ ● $y < 0$のとき x は整数または$\frac{1}{x}$が奇数 ただし $-1\times10^{100} < x \log\lvert y\rvert < 100$	$y^x = 10^{x\cdot\log y}$

関数	計算範囲		備考
&Hx	$0 \leqq x \leqq$ 2540BE3FF または FDABF41C01 $\leqq x \leqq$ FFFFFFFFFF		x は16進数での整数
POL(x,y) $(x,y \to r,\theta)$	$(x^2+y^2) < 1\times10^{100}$ $\dfrac{x}{y} < 1\times10^{100}$		$r=\sqrt{x^2+y^2}$ $\theta=\tan^{-1}\dfrac{y}{x}$
REC(r,θ) $(r,\theta \to x,y)$	$r < 1\times10^{100}$ θ の範囲は $\sin x, \cos x$ の x と同じ		$x = r\cos\theta$ $y = r\sin\theta$
NPR(n,r)	$\dfrac{n!}{(n-r)!} < 1\times10^{100}$	$0 \leqq r \leqq n \leqq 9999999999$ r,n は整数	${}_n\mathrm{P}_r$
NCR(n,r)	$\dfrac{n!}{(n-r)!r!} < 1\times10^{100}$ $n-r<r$ のとき $n-r \leqq 69$ $n-r \geqq r$ のとき $r \leqq 69$	$0 \leqq r \leqq n \leqq 9999999999$ r,n は整数	${}_n\mathrm{C}_r$
FACT x	$0 \leqq x \leqq 69$		$n!$
DEG x	$\lvert x\rvert < 1\times10^{100}$		DMS→DEG
DMS x	$\lvert x\rvert < 1\times10^{100}$		DEG→DMS

統計計算

	計算範囲		
データ	$\lvert x\rvert < 1\times10^{50}$	$1 \leqq n < 1\times10^{100}$	$\lvert y\rvert < 1\times10^{50}$
統計量	次の計算において、計算結果および途中結果の絶対値が 1×10^{100} 未満であること。 分母(除数)が0でないこと。$\sqrt{\ }$ で計算する値が負数でないこと。		

Σx　　Σx^2　　Σy　　Σy^2

$$\bar{x}=\frac{\Sigma x}{n} \qquad \bar{y}=\frac{\Sigma y}{n}$$

$$\mathrm{S}x=\sqrt{\frac{\Sigma x^2-n\bar{x}^2}{n-1}} \qquad \mathrm{S}y=\sqrt{\frac{\Sigma y^2-n\bar{y}^2}{n-1}}$$

$$\sigma x=\sqrt{\frac{\Sigma x^2-n\bar{x}^2}{n}} \qquad \sigma y=\sqrt{\frac{\Sigma y^2-n\bar{y}^2}{n}}$$

$$\mathrm{a}=\bar{y}-\mathrm{b}\bar{x} \qquad \mathrm{b}=\frac{\mathrm{S}xy}{\mathrm{S}xx}$$

$$r=\frac{\mathrm{S}xy}{\sqrt{\mathrm{S}xx\cdot\mathrm{S}yy}} \qquad \mathrm{S}xx=\Sigma x^2-\frac{(\Sigma x)^2}{n}$$

$$x'=\frac{y-\mathrm{a}}{\mathrm{b}} \qquad \mathrm{S}yy=\Sigma y^2-\frac{(\Sigma y)^2}{n}$$

$$y'=\mathrm{a}+\mathrm{b}x \qquad \mathrm{S}xy=\Sigma xy-\frac{\Sigma x\cdot\Sigma y}{n}$$

計算の誤差は原則として、10桁目に±1の誤差となります。(指数表示の場合は仮数部表示の最下位桁に±1となります。)

ただし、関数の特異点および変曲点の近傍では誤差が累積されて大きくなります。また、連続計算を行った場合はそれぞれの誤差が累積されて大きくなります。(べき乗 (y^x) のように、計算機内で連続計算を行っている場合も同様です。)

5. アフターサービスについて

修理を依頼されるときは

修理は、お買いあげの販売店に、本機の品名（ポケットコンピュータ）、形名（PC-E200）および故障状態をくわしくご説明いただいて、お申しつけください。

ご転居・ご贈答品などでお買いあげの販売店に修理を依頼することができない場合はもよりのシャープお客様ご相談窓口にお問い合わせください。

製品の保証について

①この計算機は取扱説明書の巻末に保証書がついています。保証書は販売店にて所定事項を記入いたしますので、内容をよくお読みのうえ大切に保存してください。

②保証期間はお買いあげの日から１年間です。

保証書の記載内容によりお買いあげの販売店または当社サービス会社が修理いたします。修理のときは、販売店へ持ち込みをお願いいたします。くわしくは保証書をご覧ください。

③保証期間中の修理などアフターサービスについておわかりにならない場合は、お買いあげの販売店、またはもよりのシャープお客様ご相談窓口にお問い合わせください。

④ポケットコンピュータに、制御用の外付回路を接続して使用した場合に、誤配線、誤操作、強度の外来ノイズ、あるいは許容範囲外の電圧を加えたなどによってポケットコンピュータ本体が故障・損傷した場合は、保証書に記載されている無料修理規定が適用されません（有料修理となります）ので、あらかじめご了承ください。

⑤保証期間経過後の修理については販売店にご相談ください。修理によって機能が維持できる場合はお客様のご要望により有料修理いたします。

お問い合わせは

この製品についてのご意見、ご質問はもよりのお客様ご相談窓口（技術センター）へお申しつけください。

付属の「シャープサービス・お客様ご相談窓口一覧表」のとおり、全国にお客様ご相談窓口（技術センター）を設けております。

6. 仕様

形　　　　名　PC-E200

C　P　U　CMOS　8ビットCPU（Z80A相当品）

メモリ容量　システムエリア……………………………約2.1Kバイト
　　　　　　データ専用エリア……………………………208バイト
　　　　　　プログラム・データエリア……………30435バイト

スタック　サブルーチン用　10段　ファンクション用　16段
　　　　　FOR～NEXT用　5段　データ用　8段

基本計算機能　基本計算：加減乗除算
　　　　　　　関数計算：三角関数、逆三角関数、対数、指数、角度変換、べき乗、開平計算、整数化、絶対値、符号関数、円周率、座標変換関数、その他

計算桁数　10桁（仮数部）＋2桁（指数部）

計算方法　数式通り（優先順位判別機能つき）

編集機能　カーソル左右シフト（▶、◀）
　　　　　挿入（INS）　削除（DEL、BS）
　　　　　リストアップ・ダウン（↑、↓）
　　　　　テキストエディタ、Z80機械語モニタ

周辺機器接続端子（11ピンコネクタ）
　　　　　CE-126P、CE-T800接続

メモリ保護　バッテリーバックアップ（プログラム、データなど電源オフ時内容保護）

表　　示　液晶表示　5×7ドットマトリックス表示（24桁×4行）

使用温度　0℃～40℃

電　　源　動作用電池6V⎓（DC）：単3タイプ乾電池　4本
　　　　　メモリ保護用電池3V⎓（DC）：リチウム電池〈CR2032〉1個

電池使用時間　実使用状態で連続使用　約80時間（単3タイプ乾電池SUM-3の場合）
　　　　　　　［使用温度20℃で1時間当り演算またはプログラム実行を10分間、表示状態を50分間行った場合］

注）別売の通信用ケーブルCE-T800を使用して通信を行っているときの電池使用時間は約48時間になります。
［使用温度20℃で、通信を2分間、演算またはプログラム実行を8分間、表示状態を50分間行った場合］

●電池の種類や使用方法などにより多少の変動があります。

消費電力　0.37W

外形寸法　幅215㎜×奥行100㎜×厚さ18㎜

重　　量　約280g（乾電池、リチウム電池を含む）

付属品　ハードカバー、単3乾電池　4本、リチウム電池　1個、取扱説明書、シャープサービス・お客様ご相談窓口一覧表

"故障"かな?と思ったら

本機が正しく動作しないときは、次の場合が考えられますので確認してください。

(1)電源が入らない

①電池の寿命切れの場合があります。新しい電池と交換してください。(250ページ参照)

②メモリ保護スイッチ⓯がB側になっていることが考えられます。スイッチをA側にしてください。

(2)表示が薄い、または濃すぎる

本機の右側横に付いている表示濃淡調整つまみ⓫を回して調整してください。

(3)BASICプログラムの内容が表示できない

①PASS命令でプログラムの表示を禁止していることが考えられます。パスワードを解除してください。

②プログラムが何かの原因で破壊されていることが考えられます。この場合は最初からプログラムを入れ直してください。

(4)本機が正しく動作しないとき

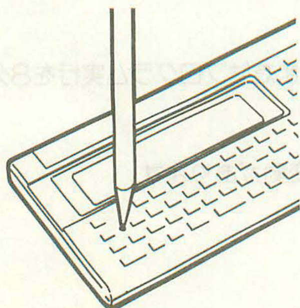
BREAK ON キーを含めたすべてのキーの機能が働かない、あるいは正しく動作しないなどの異常が発生した場合は、本機の電源を入れたままで周辺機器の接続や取りはずしを行ったか、またはプログラムのミス、あるいは強度の外来ノイズなどによる異常発生などの原因が考えられます。

また、動作用電池の交換を行った後、リセットスイッチ❺を押さなかった場合が考えられます。この場合は、次の方法でリセットしてください。

リセットのしかた

①リセットスイッチの操作

ボールペンなどで、本機左端の OFF キーの横にあるリセット(RESET)スイッチ❺を押してください。



リセットスイッチ❺を操作するときはボールペンなどで押してください。芯先の出たシャープペンシルや先の折れやすいもの、また、針など先のとがったものは使用しないでください。

リセットスイッチ❺を離すと次の画面になります。もし、違う画面になったときはもう一度リセットスイッチ❺を押してください。（画面は“メモリの内容（記憶内容）を消去してもよいか？”ときいています。）

MEMORY CLEAR O. K. ? (Y/N)

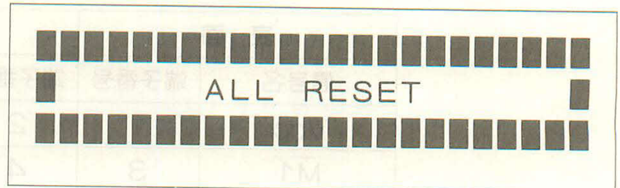
②プログラムやデータを保持したいときは、

N キー（ Y 以外のキー）を押してください。RUNモードの最初の画面になります。

●この後、プログラム実行などで再び異常が発生する場合は、③の方法を行ってください。

③プログラムやデータなどをすべて消去するときは、

上記の表示中に Y キーを押してください。記憶内容をすべて消去して、次の画面（点滅）になります。（初期設定し、記憶内容をすべて消去したことを示しています。）



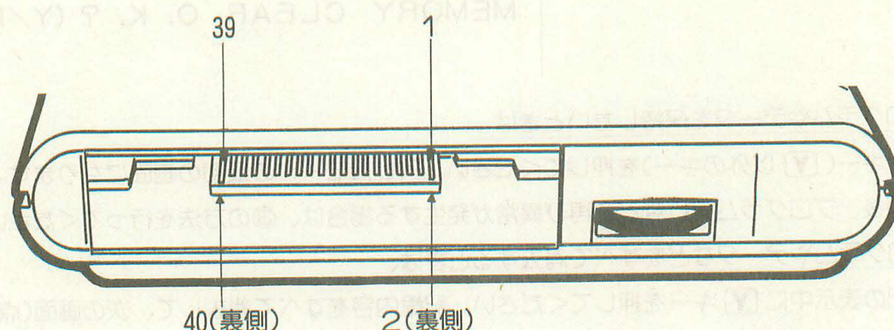
この画面は、どれかキーを押せばRUNモードの最初の画面になります。

⑸その他

⑴～⑷でもなく、また処理できない場合には、本機の故障が考えられます。このときには、「シャープサービス・お客様ご相談窓口一覧表」に記載のもよりのサービス会社へお問い合わせください。

システムバス端子信号表

表面(キー側)



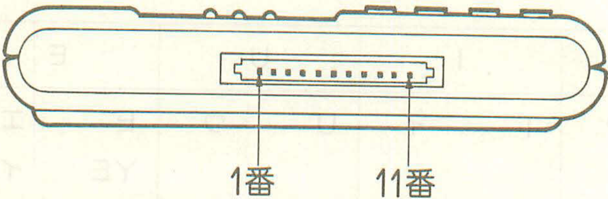
裏面

表面		裏面	
信号名	端子番号	端子番号	信号名
Vcc	1	2	Vcc
$\overline{\text{M1}}$	3	4	$\overline{\text{MREQ}}$
$\overline{\text{IORQ}}$	5	6	IORESET
$\overline{\text{WAIT}}$	7	8	$\overline{\text{INT1}}$
$\overline{\text{WR}}$	9	10	$\overline{\text{RD}}$
BNK1	11	12	BNK0
$\overline{\text{CEROM2}}$	13	14	CERAM2
D7	15	16	D6
D5	17	18	D4
D3	19	20	D2
D1	21	22	D0
A15	23	24	A14
A13	25	26	A12
A11	27	28	A10
A9	29	30	A8
A7	31	32	A6
A5	33	34	A4
A3	35	36	A2
A1	37	38	A0
GND	39	40	GND

GND：ØV　Vcc：5V±10%

注）本機内部は、C-MOS部品により構成されているため、端子に本機の許容範囲（ØV～+5V±10%）を超える電圧が加えられると、本機内部が壊れることがありますので注意してください。

周辺機器接続端子信号表



端子番号	端子名	本機から見た信号方向	機能概要
1	MTOUT2	入力	●オプション機器から本機のブザーを駆動するための信号。
2	電源(＋)	——	●システム電源。
3	電源(－)	——	●グランド。システム全体の信号の基準を定める。
4	BUSY	出力	●周辺機器とデータ交換を行うための制御信号。 ●ＳＩＯのRS信号。
5	Dout	出力	●プリンタへのデータ出力。 ●ＳＩＯの通信のとき、CE-T800のRS-232Cレベル発生回路を動作させる。
6	Xin	入力	●カセットからのデータ入力。 ●ＳＩＯのRD信号。
7	Xout	出力	●カセットへのデータ出力。 ●ＳＩＯのSD信号。
8	Din	入力	●プリンタの異常(紙づまり、ローバッテリー) 情報の入力。
9	ACK	入力	●周辺機器とデータ交換を行うための制御信号。 ●ＳＩＯのCS信号。
10			使用していません。
11			使用していません。

電源(－)：ØV　　電源(＋)：５V±10%

注）本機内部は、C-MOS部品により構成されているため、端子に本機の許容範囲（ØV～＋５V±10%)を超える電圧が加えられると、本機内部が壊れることがありますので注意してください。

注）BUSY（４番端子）とDout（５番端子）の出力信号は、ハイレベル信号（電源(＋)のレベル）は出力されますが、ローレベル信号は出力されません。したがって、これらの信号を外部から使用する場合は、100KΩ程度の抵抗で、電源(－)側に接続する（プルダウンする）必要があります。

ローマ字→カナ変換表

	A		I		U		E		O	
ア行	A	ア	I	イ	U	ウ	E YE	エ イエ	O	オ
カ行	KA CA QA KYA	カ カ クァ キャ	KI QI KYI	キ クィ キィ	KU CU QU KYU	ク ク ク キュ	KE QE KYE	ケ クェ キェ	KO CO QO KYO	コ コ クォ キョ
サ行	SA SHA SYA	サ シャ シャ	SI SHI SYI	シ シ シィ	SU SHU SYU	ス シュ シュ	SE SHE SYE	セ シェ シェ	SO SHO SYO	ソ ショ ショ
タ行	TA TSA CHA TYA CYA	タ ツァ チャ チャ チャ	TI TSI CHI TYI CYI	チ ツィ チ チィ チィ	TU TSU CHU TYU CYU	ツ ツ チュ チュ チュ	TE TSE CHE TYE CYE	テ ツェ チェ チェ チェ	TO TSO CHO TYO CYO	ト ツォ チョ チョ チョ
ナ行	NA NYA	ナ ニャ	NI NYI	ニ ニィ	NU NYU	ヌ ニュ	NE NYE	ネ ニェ	NO NYO	ノ ニョ
ハ行	HA FA HYA	ハ ファ ヒャ	HI FI HYI	ヒ フィ ヒィ	HU FU HYU	フ フ ヒュ	HE FE HYE	ヘ フェ ヒェ	HO FO HYO	ホ フォ ヒョ
マ行	MA MYA	マ ミャ	MI MYI	ミ ミィ	MU MYU	ム ミュ	ME MYE	メ ミェ	MO MYO	モ ミョ
ヤ行	YA	ヤ	YI	イ	YU	ユ			YO	ヨ
ラ行	RA LA RYA LYA	ラ ラ リャ リャ	RI LI RYI LYI	リ リ リィ リィ	RU LU RYU LYU	ル ル リュ リュ	RE LE RYE LYE	レ レ リェ リェ	RO LO RYO LYO	ロ ロ リョ リョ
ワ行	WA	ワ							WO	ヲ
ン	N	N SHIFT + U			M					

	A		I		U		E		O	
ア行	VA	ヴァ	VI	ヴィ	VU	ヴ	VE	ヴェ	VO	ヴォ
ガ行	GA GYA	ガ ギャ	GI GYI	ギ ギィ	GU GYU	グ ギュ	GE GYE	ゲ ギェ	GO GYO	ゴ ギョ
ザ行	ZA JA JYA ZYA	ザ ジャ ジャ ジャ	ZI JI JYI ZYI	ジ ジ ジィ ジィ	ZU JU JYU ZYU	ズ ジュ ジュ ジュ	ZE JE JYE ZYE	ゼ ジェ ジェ ジェ	ZO JO JYO ZYO	ゾ ジョ ジョ ジョ
ダ行	DA DHA DYA	ダ デャ ヂャ	DI DHI DYI	ヂ ディ ヂィ	DU DHU DYU	ヅ デュ ヂュ	DE DHE DYE	デ デェ ヂェ	DO DHO DYO	ド デョ ヂョ
バ行	BA BYA	バ ビャ	BI BYI	ビ ビィ	BU BYU	ブ ビュ	BE BYE	ベ ビェ	BO BYO	ボ ビョ
パ行	PA PYA	パ ピャ	PI PYI	ピ ピィ	PU PYU	プ ピュ	PE PYE	ペ ピェ	PO PYO	ポ ピョ

①「ン」の入力

「ン」は普通「N」と入力します。

ただし、「N」の次に母音(A、I、U、E、O)および「Y」がくるとき、または後に文字がこないときは「N SHIFT + Ù」と押します。

なお、「N」の代わりに「M」を用いることもできます。

(例) シンユウ――→SIN SHIFT + Ù YUU

シンニュウ――→SINNYUU

シンブン――→SIMBUN SHIFT + Ù

②「ッ」の入力

「ッ」は子音を重ねて入力します。ただし、「N」や「M」を重ねて入力したときは「ン」が入ります。

(例) キップ――→KIPPU　　セット――→SETTO

③小さい文字の単独入力

小さい文字(ァ、ィ、ゥ、ェ、ォ、ッ、ャ、ュ、ョ)を入力するときは 小文字 CAPS キーを押してからそれぞれの文字を入れます。

(例) ティーカップ――→TE 小文字 CAPS I－KAPPU

キャラクタ・コード表

下位桁	上位桁	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
	16進数	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	ヌル		スペース	Ø	@	P	'	p		⊥		ー	タ	ミ	＝	×
1	1			!	1	A	Q	a	q	▁	┬	。	ア	チ	ム	╞	円
2	2			"	2	B	R	b	r	▂	┤	「	イ	ツ	メ	╪	年
3	3			#	3	C	S	c	s	▃	├	」	ウ	テ	モ	╡	月
4	4			$	4	D	T	d	t	▄	‾	、	エ	ト	ヤ	◢	日
5	5			%	5	E	U	e	u	▅	─	・	オ	ナ	ユ	◣	時
6	6			&	6	F	V	f	v	▆	▏	ヲ	カ	ニ	ヨ	◥	分
7	7			'	7	G	W	g	w	▇	▕	ァ	キ	ヌ	ラ	◤	秒
8	8			(	8	H	X	h	x	▏	┌	ィ	ク	ネ	リ	♠	
9	9			)	9	I	Y	i	y	▎	┐	ゥ	ケ	ノ	ル	♥	
10	A			*	:	J	Z	j	z	▍	└	ェ	コ	ハ	レ	♦	
11	B			+	;	K	[	k	{	▌	┘	ォ	サ	ヒ	ロ	♣	
12	C			,	<	L	¥	l	¦	▋	╭	ャ	シ	フ	ワ	●	
13	D			−	=	M	]	m	}	▊	╮	ュ	ス	ヘ	ン	○	
14	E			.	>	N	^	n	~	▉	╰	ョ	セ	ホ	゛	／	
15	F			／	?	O	_	o		┼	╯	ッ	ソ	マ	゜	＼	

キャラクタコードは次のように表わします。

〈例〉　"*" のコード

16進数　&H2A

10進数　42（32+10）

〈ご注意〉

☆CHR$命令により、本機でキャラクタを表示させる場合

- 表中のコードØ（&HØØ）のキャラクタはヌル（Null：何もない状態）です。
  したがって何も表示されません。
- キャラクタが記載されていない部分はスペース（空）になります。

☆CHR$命令により、CE-126Pでキャラクタをプリントさせる場合

- キャラクタが記載されていない部分はスペース（空）になります。
- 次のコードはスペースになります。
  129(&H81)〜159(&H9F)、224(&HE0)〜231(&HE7)、236(&HEC)〜240(&HF0)、245(&HF5)〜247(&HF7)

エラーコード表

エラーコード	内　容
10	文法的に実行できない場合のエラー。
12	プログラムの中だけで実行できる命令をマニュアル操作で実行しようとした。または、逆にマニュアル操作だけで実行できる命令をプログラムで実行しようとした。 ＰＲＯモードでしか使えない命令(ＬＩＳＴ、ＲＥＮＵＭなど)をＲＵＮモードで使おうとした。または、逆にＲＵＮモードでしか使えない命令をＰＲＯモードで使おうとした。
13	ＣＯＮＴ命令でプログラムの再実行ができない。
14	ＢＡＳＩＣプログラムがないときに、パスワードを設定しようとした。
15	ＣＳＡＶＥ Ｍ命令で、アドレスの指定が逆。(開始アドレスよりも、終了アドレスのほうが小さくなっている。)
20	計算結果が１×10^{100}以上になった。
21	除数が０の除算を実行した。
22	不合理な演算を行った。（例　ＬＯＧ−３） または、計算範囲外の値の関数計算を行った。（例　ＡＳＮ　２）
30	すでに宣言されている配列変数名を再度宣言している。
31	ＤＩＭ命令で宣言していない配列変数を使用している。
32	配列変数の添字がＤＩＭ命令で宣言した大きさを超えている。
33	指定している数値が規定の範囲から外れている。
40	ＧＯＴＯ、ＬＩＳＴ、ＲＥＮＵＭなどの命令で指定した行番号やラベルが存在しない。
41	行番号(ラインナンバー)が1〜65279の範囲外になっている。
43	ＲＥＮＵＭ命令の指定に不都合がある。 （指定した開始行以降の行番号の付け替えを行うと、開始行よりも小さい番号の行（行番号の付け替えをしない行）と混ざってしまうような指定になっている。）
44	ＬＬＩＳＴ、ＤＥＬＥＴＥ命令で指定した開始行と終了行の大きさが逆になっている。（終了行よりも開始行が大きくなっている。）
50	ＧＯＳＵＢ(10段)、ＦＯＲ〜ＮＥＸＴ(５段)での段数オーバー。
51	ＲＥＴＵＲＮ文に対するＧＯＳＵＢ文がない。
52	ＮＥＸＴ文に対するＦＯＲ文がない。
53	ＲＥＡＤ文に対するＤＡＴＡ文がない。
54	データバッファ(８段)または、ファンクションバッファ(16段)の段数オーバー。
55	文字記憶エリア(255文字)の容量を超えた。 １行の長さが255バイトを超えた。
60	プログラムおよび変数が大きすぎて、プログラム・データエリアの容量を超えた。
70	ＰＲＩＮＴ、ＬＰＲＩＮＴ命令で数値データを表示または印字するとき、ＵＳＩＮＧ命令で指定されたフォーマットで表示または印字できない。
71	ＵＳＩＮＧ命令の指定が正しくない。(" "で囲んで指定した内容が正しくない。)

エラーコード	内　　　　　容
80	カセットテープに対するチェックサムエラー、ＳＩＯに対するリードイン（読み込み）エラー
81	カセットテープ、ＳＩＯ、ミニＩ／Ｏなどのタイムアウトエラー（プログラムやデータの入出力で、規定の待ち時間を超えた。）
82	カセットテープ等の内容の照合で、内容の不一致がある。
83	ＩＮＰＵＴ＃命令で指定している変数の型が、読み込もうとしているデータの型と一致していない。
84	プリンタ関連のエラー。
85	ＰＲＩＮＴ＃、ＩＮＰＵＴ＃命令でデータの入出力を行うとき、相手の装置（デバイス）がオープンしていない。
86	１つの装置（デバイス）がオープンしているときに、ほかの装置をオープンしようとした。
87	ファイルのデータを最後まで読み込んだ後、さらにデータを読み込もうとした。
90	数値変数に文字、文字変数に数値を代入しようとした。また、ＳＩＮ Ａ＄ のように数値を扱う関数に文字変数を指定したなど、変数名の不適合。
91	固定変数において、数値が入っている変数を文字変数として使用したり、文字が入っている変数を数値変数として使用した。
92	パスワードが一致していない。
93	パスワードが設定されているときに、マニュアルでＭＯＮ、ＰＥＥＫ、ＰＯＫＥなどの命令を実行しようとした。
94	指定されたファイルが存在しない。
95	ファイル名の指定が正しくない。
96	ＢＡＳＩＣモードでＴＥＸＴファイルを読み込もうとした。また逆にＴＥＸＴモードでＢＡＳＩＣファイルを読み込もうとした。
97	ファイルの数が255を超えた。

保証書(保証規定)

本書は、本書記載内容で無料修理をさせて頂くことをお約束するものです。
保証期間中に故障が発生した場合は、製品と本書をご持参、ご提示の上、お買上げの販売店にご依頼ください。
お買上げ年月日、販売店名など記入もれがありますと無効となります。必ずご確認いただき、記入のない場合はお買上げの販売店にお申し出ください。
ご転居・ご贈答品などでお買上げ販売店に修理をご依頼できない場合は、製品に同梱しております、「シャープサービス・お客様ご相談窓口一覧表」をご覧の上、もよりのサービス会社へご持参ください。
本書は再発行いたしません。たいせつに保管してください。

〈無料修理規定〉

1．取扱説明書・本体注意ラベルなどの注意書にしたがった正常な使用状態で、保証期間内に故障した場合には、お買上げ販売店、または当社サービス会社が無料修理いたします。
2．保証期間内でも次の場合には有料修理となります。
(イ)本書のご提示がない場合。
(ロ)本書にお買上げ年月日・お客様名・販売店名の記入がない場合、または字句を書き換えられた場合。
(ハ)使用上の誤り、または不当な修理や改造による故障・損傷。
(ニ)お買上げ後に落とされた場合などによる故障・損傷。
(ホ)火災・公害および地震・風水害その他天災地変など、外部に要因がある故障・損傷。
(ヘ)電池の液もれによる故障・損傷。
(ト)消耗品（乾電池、リチウム電池）が損耗し取り替えを要する場合。
3．本書は日本国内においてのみ有効です。
(THIS WARRANTY CARD IS ONLY VALID FOR SERVICE IN JAPAN.)

★この保証書は本書に明示した期間・条件のもとにおいて無料修理をお約束するものです。したがいましてこの保証書によってお客様の法律上の権利を制限するものではありませんので、保証期間経過後の修理などにつきましておわかりにならない場合はお買上げの販売店、またはシャープお客様ご相談窓口にお問い合せください。

修理メモ

シャープポケットコンピュータ保証書 持込修理
(WARRANTY CARD)

品　　名	ポケットコンピュータ
形　　名	PC-E200
保証期間 (VALIDITY)	お買上げ日より本体１年間 (FULL 1 YEAR AFTER PURCHASE)
お買上げ日 (PURCHASE)	昭和　年　月　日

お客様	ご芳名	様
	ご住所	〒
	電話番号	(　　)　　－
取扱販売店名・住所・電話番号		

ショッカー
(株)十字屋船橋店
船橋市本町１－７－６
TEL 0474－31－1191(代)

印

シャープ株式会社
〒545 大阪市阿倍野区長池町22番22号
電話 (06) 621－1221番

シャープ株式会社

本　　　　　社　〒545 大阪市阿倍野区長池町22-22
電話（06）621-1221（大代表）

情報システム　〒639-11　奈良県大和郡山市美濃庄町492
事業本部　電話（07435）3-5521（大代表）

修理などアフターサービスについてのお問い合わせは下記へ……

シャープお客様ご相談窓口（技術センター）

札幌　技術センター	〶(063) 札幌市西区二十四軒1条7丁目3番17号	電話(011)641－0751
仙台　技術センター	〶(983) 仙台市卸町東3丁目1番27号	電話(022)288－9161
岩手　技術センター	〶(028-36) 岩手県柴波郡矢巾町流通センター南3丁目1番1号	電話(0196)38－6085
宇都宮技術センター	〶(320) 宇都宮市不動前4丁目2番41号	電話(0286)34－0256
新潟　技術センター	〶(950) 新潟市上所中1丁目7番21号	電話(025)284－6023
松本　技術センター	〶(399-65) 松本市芳野8番14号	電話(0263)27－1636
東京第1技術センター	〶(130) 東京都墨田区石原2丁目12番3号	電話 (03)624－7475
東京第2技術センター	〶(164) 東京都中野区南台3丁目45番13号	電話 (03)382－9161
東京第3技術センター	〶(143) 東京都大田区南馬込1丁目5番15号	電話 (03)777－8851
大宮　技術センター	〶(330) 大宮市宮原町2丁目107番2号	電話(0486)66－7220
千葉　技術センター	〶(280) 千葉市南町1丁目5番20号	電話(0472)65－6110
横浜　技術センター	〶(235) 横浜市磯子区中原1丁目2番23号	電話(045)753－9583
名古屋技術センター	〶(454) 名古屋市中川区山王3丁目5番5号	電話(052)332－2671
岐阜　技術センター	〶(500) 岐阜市六条南3丁目12番9号	電話(0582)74－7996
金沢　技術センター	〶(921) 石川県石川郡野々市町字御経塚町1096の1	電話(0762)49－9033
大阪第1技術センター	〶(556) 大阪市浪速区恵美須西1丁目2番9号	電話 (06)644－1191
大阪第2技術センター	〶(547) 大阪市平野区加美南4丁目3番41号	電話 (06)794－2910
神戸　技術センター	〶(658) 神戸市東灘区魚崎北町1丁目6番18号	電話(078)452－1762
京都　技術センター	〶(601) 京都市南区上鳥羽管田町48番地	電話(075)681－9551
高松　技術センター	〶(760) 高松市木太町1861番3号	電話(0878)35－5525
岡山　技術センター	〶(700) 岡山市米倉字東ノ町66の2	電話(0862)43－5660
広島　技術センター	〶(731-01) 広島市安佐南区西原2丁目13番4号	電話(082)874－6100
山口　技術センター	〶(745) 徳山市新地1番地	電話(0834)32－3839
福岡　技術センター	〶(816) 福岡市博多区井相田2丁目12番1号	電話(092)572－2617
北九州技術センター	〶(803) 北九州市小倉北区大手町6番12号	電話(093)592－6510
鹿児島技術センター	〶(890) 鹿児島市鴨池新町12番1号	電話(0992)59－0628

シャープ株式会社

本　　　　　　　社　　〒545 大阪市阿倍野区長池町22番22号
電話(06)621－1221(大代表)

情報システム事業本部　〒639-11 奈良県大和郡山市美濃庄町492番地
電話(07435)3－5521(大代表)

8B9.1KS (TINSJ1168ECZ